ESCALA

Virtual I/O Server

REFERENCE
86 A1 81FA 01

ESCALA

Virtual I/O Server

Hardware

May 2009

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

REFERENCE
86 A1 81FA 01

Printed in France

Trademarks and Acknowledgements

The information in this document is subject to change without notice. Bull will not be liable for errors contained herein, or for incidental or consequential damages in connection with the use of this material.

Contents

Virtual I/O Server

Manage the Virtual I/O Server and client logical partitions using the Hardware Management Console (HMC) and the Virtual I/O Server command-line interface.

The PowerVM™ Editions feature includes the installation media for the Virtual I/O Server software. The Virtual I/O Server facilitates the sharing of physical I/O resources between client logical partitions within the server.

When you install the Virtual I/O Server in a logical partition on a system that is managed by the HMC, you can use the HMC and the Virtual I/O Server command-line interface to manage the Virtual I/O Server and client logical partitions.

When you install the Virtual I/O Server on a managed system and there is no HMC attached to the managed system when you install the Virtual I/O Server, then the Virtual I/O Server logical partition becomes the management partition. The management partition provides the Integrated Virtualization Manager Web-based system management interface and a command-line interface that you can use to manage the system.

Related information

PowerVM Information Roadmap

Integrated Virtualization Manager

Virtual I/O Server and Integrated Virtualization Manager commands

What’s new in Virtual I/O Server

Read about new or significantly changed information in Virtual I/O Server since the previous update of this topic collection.

May 2009

The following updates have been made to the content:

- You can use the HMC graphical interface, version 7 release 3.4.2 or later to complete many virtual storage and networking tasks for a Virtual I/O Server. In previous versions, you were required to complete these tasks using a Virtual I/O Server command-line interface. The following tasks are updated to include links to the HMC graphical interface instructions, when appropriate.
  - Creating a virtual target device on a Virtual I/O Server
  - Creating logical volume storage pools on a Virtual I/O Server
  - Creating file storage pools on a Virtual I/O Server
  - Creating the virtual media repository on a Virtual I/O Server
  - Creating volume groups and logical volumes on a Virtual I/O Server
  - Configuring a shared Ethernet adapter
  - Assigning the virtual fibre channel adapter to a physical fibre channel adapter
  - Increasing virtual SCSI device capacity
- Logical partitions can share the memory in the shared memory pool by using the PowerVM Active Memory™ Sharing technology (or shared memory). A Virtual I/O Server (VIOS) logical partition that is assigned to the shared memory pool (hereafter referred to as a paging VIOS partition) provides access to the paging space devices for the logical partitions that are assigned to the shared memory pool. The following information is new for shared memory:

- – Shared memory
- – Paging VIOS partition
- – Paging VIOS partition requirements

For more information about what’s new for shared memory, see What’s new in Logical partitioning.

November 2008

- With N_Port ID Virtualization (NPIV) and virtual fibre channel adapters, you can configure the managed system so that multiple logical partitions can access independent physical storage through the same physical fibre channel adapter. The following information is new for virtual fibre channel adapters:
  - – Virtual fibre channel
  - – Configuring a virtual fibre channel adapter
  - – “Assigning the virtual fibre channel adapter to a physical fibre channel adapter” on page 108
  - – Managing virtual Fibre Channel on the Integrated Virtualization Manager
  - – Redundancy configuration using virtual fibre channel adapters

Virtual I/O Server overview

Learn the concepts of the Virtual I/O Server and its primary components.

The Virtual I/O Server is software that is located in a logical partition. This software facilitates the sharing of physical I/O resources between client logical partitions within the server. The Virtual I/O Server provides virtual SCSI target, virtual fibre channel, Shared Ethernet Adapter, and PowerVM Active Memory Sharing capability to client logical partitions within the system. As a result, client logical partitions can share SCSI devices, fibre channel adapters, Ethernet adapters, and expand the amount of memory available to logical partitions using paging space devices. The Virtual I/O Server software requires that the logical partition be dedicated solely for its use.

The Virtual I/O Server is part of the PowerVM Editions hardware feature.

Using the Virtual I/O Server facilitates the following functions:

- Sharing of physical resources between logical partitions on the system
- Creating logical partitions without requiring additional physical I/O resources
- Creating more logical partitions than there are I/O slots or physical devices available with the ability for logical partitions to have dedicated I/O, virtual I/O, or both
- Maximizing use of physical resources on the system
- Helping to reduce the Storage Area Network (SAN) infrastructure

Related information

Virtual I/O Server and Integrated Virtualization Manager commands

Operating system support for VIOS client logical partitions

The Virtual I/O Server supports client logical partitions that run the following operating systems on the following POWER6™ processor-based servers.

Table 1. Operating system support for Virtual I/O Server client logical partitions

Operating system	POWER6 processor-based servers
AIX® 5.3 or later	All POWER6 processor-based servers
IBM® i 6.1 or later	All POWER6 processor-based servers

Table 1. Operating system support for Virtual I/O Server client logical partitions (continued)

Operating system	POWER6 processor-based servers
SUSE Linux® Enterprise Server 10 Service Pack 2 or later	• 9119-FHA • 9125-F2A
SUSE Linux Enterprise Server 10 Service Pack 1	• 8203-E4A • 8204-E8A • 9117-MMA • 9406-MMA • 9407-M15 • 9408-M25 • 9409-M50
Red Hat Enterprise Linux version 5.2	• 9119-FHA • 9125-F2A
Red Hat Enterprise Linux version 5.1	• 8203-E4A • 8204-E8A • 9117-MMA • 9406-MMA • 9407-M15 • 9408-M25 • 9409-M50
Red Hat Enterprise Linux version 4.7	9119-FHA
Red Hat Enterprise Linux version 4.6	9125-F2A
Red Hat Enterprise Linux version 4.5	• 8203-E4A • 8204-E8A • 9117-MMA • 9406-MMA • 9407-M15 • 9408-M25 • 9409-M50

Components of the Virtual I/O Server

This topic provides a brief overview of virtual SCSI, virtual networking, and the Integrated Virtualization Manager.

For the most recent information about devices that are supported on the Virtual I/O Server and to download Virtual I/O Server fixes and updates, see the Virtual I/O Server Web site.

The Virtual I/O Server comprises the following primary components:

- Virtual SCSI
- Virtual networking
- Integrated Virtualization Manager

The following sections provide a brief overview of each of these components.

Virtual SCSI

Physical adapters with attached disks or optical devices on the Virtual I/O Server logical partition can be shared by one or more client logical partitions. The Virtual I/O Server offers a local storage subsystem that provides standard SCSI-compliant logical unit numbers (LUNs). The Virtual I/O Server can export a pool of heterogeneous physical storage as a homogeneous pool of block storage in the form of SCSI disks.

Unlike typical storage subsystems that are physically located in the storage area network (SAN), the SCSI devices exported by the Virtual I/O Server are limited to the domain within the server. Although the SCSI LUNs are SCSI compliant, they might not meet the needs of all applications, particularly those that exist in a distributed environment.

The following SCSI peripheral-device types are supported:

- Disk backed up by a logical volume
- Disk backed up by a physical volume
- Disk backed up by a file
- Optical devices (DVD-RAM and DVD-ROM)
- Optical devices backed up by files
- Tape devices

Virtual networking

Virtual I/O Server provides the following virtual networking technologies.

Table 2. Virtual networking technologies on the Virtual I/O Server

Virtual networking technology	Description
Shared Ethernet Adapter	A Shared Ethernet Adapter is a layer-2 Ethernet bridge that connects physical and virtual networks together. It allows logical partitions on the virtual local area network (VLAN) to share access to a physical Ethernet adapter and to communicate with systems outside the server. Using a Shared Ethernet Adapter, logical partitions on the internal VLAN can share the VLAN with stand-alone servers. On POWER6 processor-based systems, you can assign a logical host Ethernet port, of a logical host Ethernet adapter, which is sometimes referred to as Integrated Virtual Ethernet, as the real adapter of a Shared Ethernet Adapter. A Host Ethernet Adapter is a physical Ethernet adapter that is integrated directly into the GX+ bus on a managed system. Host Ethernet Adapters offer high throughput, low latency, and virtualization support for Ethernet connections. The Shared Ethernet Adapter on the Virtual I/O Server supports IPv6. IPv6 is the next generation of Internet Protocol and is gradually replacing the current Internet standard, Internet Protocol version 4 (IPv4). The key IPv6 enhancement is the expansion of the IP address space from 32 bits to 128 bits, providing virtually unlimited, unique IP addresses.

Table 2. Virtual networking technologies on the Virtual I/O Server (continued)

Virtual networking technology	Description
Shared Ethernet Adapter failover	Shared Ethernet Adapter failover provides redundancy by configuring a backup Shared Ethernet Adapter on a different Virtual I/O Server logical partition that can be used if the primary Shared Ethernet Adapter fails. The network connectivity in the client logical partitions continues without disruption.
Link Aggregation (or EtherChannel)	A Link Aggregation (or EtherChannel) device is a network port-aggregation technology that allows several Ethernet adapters to be aggregated. The adapters can then act as a single Ethernet device. Link Aggregation helps provide more throughput over a single IP address than would be possible with a single Ethernet adapter.
Virtual local area networks (VLAN)	VLAN allows the physical network to be logically segmented.

Integrated Virtualization Manager

The Integrated Virtualization Manager provides a browser-based interface and a command-line interface that you can use to manage some servers that use the Virtual I/O Server. On the managed system, you can create logical partitions, manage the virtual storage and virtual Ethernet, and view service information related to the server. The Integrated Virtualization Manager is packaged with the Virtual I/O Server, but it is activated and usable only on certain platforms and where no Hardware Management Console (HMC) is present.

Virtual fibre channel

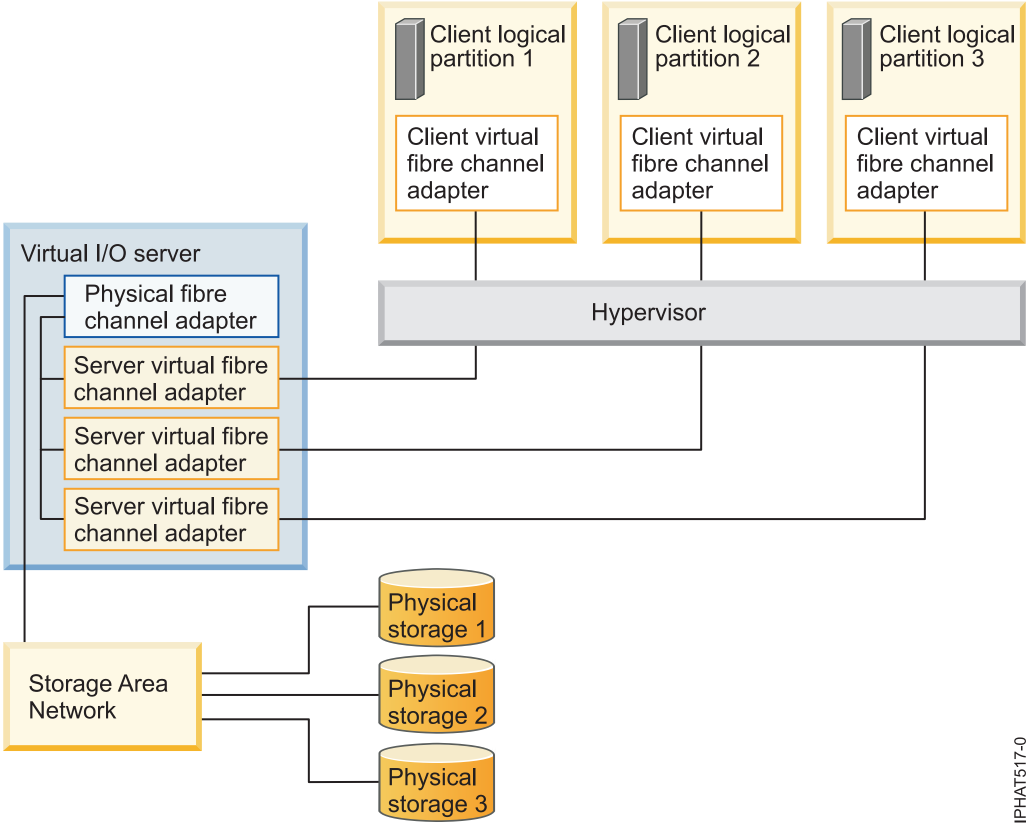
With N_Port ID Virtualization (NPIV), you can configure the managed system so that multiple logical partitions can access independent physical storage through the same physical fibre channel adapter.

To access physical storage in a typical storage area network (SAN) that uses fibre channel, the physical storage is mapped to logical units (LUNs) and the LUNs are mapped to the ports of physical fibre channel adapters. Each physical port on each physical fibre channel adapter is identified using one worldwide port name (WWPN).

NPIV is a standard technology for fibre channel networks that enables you to connect multiple logical partitions to one physical port of a physical fibre channel adapter. Each logical partition is identified by a unique WWPN, which means that you can connect each logical partition to independent physical storage on a SAN.

To enable NPIV on the managed system, you must create a Virtual I/O Server logical partition (version 2.1, or later) that provides virtual resources to client logical partitions. You assign the physical fibre channel adapters (that support NPIV) to the Virtual I/O Server logical partition. Then, you connect virtual fibre channel adapters on the client logical partitions to virtual fibre channel adapters on the Virtual I/O Server logical partition. A *virtual fibre channel adapter* is a virtual adapter that provides client logical partitions with a fibre channel connection to a storage area network through the Virtual I/O Server logical partition. The Virtual I/O Server logical partition provides the connection between the virtual fibre channel adapters on the Virtual I/O Server logical partition and the physical fibre channel adapters on the managed system.

The following figure shows a managed system configured to use NPIV.


Client logical partition 1
Client virtual fibre channel adapter
Client logical partition 2
Client virtual fibre channel adapter
Client logical partition 3
Client virtual fibre channel adapter
Virtual I/O server
Physical fibre channel adapter
Server virtual fibre channel adapter
Server virtual fibre channel adapter
Server virtual fibre channel adapter
Hypervisor
Physical storage 1
Physical storage 2
Physical storage 3
Storage Area Network
IPHAT517-0

The figure shows the following connections:

- A storage area network (SAN) connects three units of physical storage to a physical fibre channel adapter that is located on the managed system. The physical fibre channel adapter is assigned to the Virtual I/O Server and supports NPIV.
- The physical fibre channel adapter connects to three virtual fibre channel adapters on the Virtual I/O Server. All three virtual fibre channel adapters on the Virtual I/O Server connect to the same physical port on the physical fibre channel adapter.
- Each virtual fibre channel adapter on the Virtual I/O Server connects to one virtual fibre channel adapter on a client logical partition. Each virtual fibre channel adapter on each client logical partition receives a pair of unique WWPNs. The client logical partition uses one WWPN to log into the SAN at any given time. The other WWPN is used when you move the client logical partition to another managed system.

Using their unique WWPNs and the virtual fibre channel connections to the physical fibre channel adapter, the operating systems that run in the client logical partitions discover, instantiate, and manage their physical storage located on the SAN. In the previous figure, Client logical partition 1 accesses Physical storage 1, Client logical partition 2 accesses Physical storage 2, and Client logical partition 3 accesses Physical storage 3. The Virtual I/O Server cannot access and does not emulate the physical storage to which the client logical partitions have access. The Virtual I/O Server provides the client logical partitions with a connection to the physical fibre channel adapters on the managed system.

There is always a one-to-one relationship between virtual fibre channel adapters on the client logical partitions and the virtual fibre channel adapters on the Virtual I/O Server logical partition. That is, each virtual fibre channel adapter on a client logical partition must connect to only one virtual fibre channel adapter on the Virtual I/O Server logical partition, and each virtual fibre channel on the Virtual I/O Server logical partition must connect to only one virtual fibre channel adapter on a client logical partition.

Using SAN tools, you can zone and mask LUNs that include WWPNs that are assigned to virtual fibre channel adapters on client logical partitions. The SAN uses WWPNs that are assigned to virtual fibre channel adapters on client logical partitions the same way it uses WWPNs that are assigned to physical ports.

You can configure virtual fibre channel adapters on client logical partitions that run the following operating systems:

- AIX version 6.1 Technology Level 2, or later
- AIX 5.3 Technology Level 9
- SUSE Linux Enterprise Server 11, or later

Related concepts

"Redundancy configuration using virtual fibre channel adapters" on page 74
Redundancy configurations help protect your network from physical adapter failures as well as Virtual I/O Server failures.

Virtual fibre channel for HMC-managed systems

On systems that are managed by the Hardware Management Console (HMC), you can dynamically add and remove virtual fibre channel adapters to and from the Virtual I/O Server logical partition and each client logical partition. You can also view information about the virtual and physical fibre channel adapters and the worldwide port names (WWPNs) by using Virtual I/O Server commands.

To enable N_Port ID Virtualization (NPIV) on the managed system, you create the required virtual fibre channel adapters and connections as follows:

- You use the HMC to create virtual fibre channel adapters on the Virtual I/O Server logical partition and associate them with virtual fibre channel adapters on the client logical partitions.
- You use the HMC to create virtual fibre channel adapters on each client logical partition and associate them with virtual fibre channel adapters on the Virtual I/O Server logical partition. When you create a virtual fibre channel adapter on a client logical partition, the HMC generates a pair of unique WWPNs for the client virtual fibre channel adapter.
- You connect the virtual fibre channel adapters on the Virtual I/O Server to the physical ports of the physical fibre channel adapter by running the vfcmap command on the Virtual I/O Server.

The HMC generates WWPNs based on the range of names available for use with the prefix in the vital product data on the managed system. This 6–digit prefix comes with the purchase of the managed system and includes 32 000 pairs of WWPNs. When you remove a virtual fibre channel adapter from a client logical partition, the hypervisor deletes the WWPNs that are assigned to the virtual fibre channel adapter on the client logical partition. The HMC does not reuse the deleted WWPNs when generating WWPNs for virtual fibre channel adapters in the future. If you run out of WWPNs, you must obtain an activation code that includes another prefix with another 32 000 pairs of WWPNs.

To avoid configuring the physical fibre channel adapter to be a single point of failure for the connection between the client logical partition and its physical storage on the SAN, do not connect two virtual fibre channel adapters from the same client logical partition to the same physical fibre channel adapter. Instead, connect each virtual fibre channel adapter to a different physical fibre channel adapter.

You can dynamically add and remove virtual fibre channel adapters to and from the Virtual I/O Server logical partition and to and from client logical partitions.

Table 3. Dynamic logical partitioning tasks and results for virtual fibre channel adapters

Dynamically add or remove virtual fibre channel adapter	To or from a client logical partition or a Virtual I/O Server logical partition	Result
Add a virtual fibre channel adapter	To a client logical partition	The HMC generates the a pair of unique WWPNs for the client virtual fibre channel adapter.
Add a virtual fibre channel adapter	To a Virtual I/O Server logical partition	You need to connect the virtual fibre channel adapter to a physical port on a physical fibre channel adapter.
Remove a virtual fibre channel adapter	From a client logical partition	• The hypervisor deletes the WWPNs and does not reuse them. • You must either remove the associated virtual fibre channel adapter from the Virtual I/O Server, or associate it with another virtual fibre channel adapter on a client logical partition.
Remove a virtual fibre channel adapter	From a Virtual I/O Server logical partition	• The Virtual I/O Server removes the connection to the physical port on the physical fibre channel adapter. • You must either remove the associated virtual fibre channel adapter from the client logical partition, or associate it with another virtual fibre channel adapter on the Virtual I/O Server logical partition.

The following table lists the Virtual I/O Server commands that you can run to view information about the fibre channel adapters.

Table 4. Virtual I/O Server commands that display information about fibre channel adapters

Virtual I/O Server command	Information displayed by command
lsmap	• Displays the virtual fibre channel adapters on the Virtual I/O Server that are connected to the physical fibre channel adapter • Displays attributes of the virtual fibre channel adapters on the client logical partitions that are associated with the virtual fibre channel adapters on the Virtual I/O Server that are connected to the physical fibre channel adapter
lsnports	Displays information about the physical ports on the physical fibre channel adapters that support NPIV, such as: • The name and location code of the physical port • The number of available physical ports • The total number of WWPNs that the physical port can support • Whether the switches, to which the physical fibre channel adapters are cabled, support NPIV

You can also run the lshwres command on the HMC to display the remaining number of WWPNs and to display the prefix that is currently used to generate the WWPNs.

Virtual fibre channel on IVM-managed systems

On systems that are managed by the Integrated Virtualization Manager (IVM), you can dynamically add and remove worldwide port names (WWPNs) to and from logical partitions, and you can dynamically change the physical ports to which the WWPNs are assigned. You can also view information about the virtual and physical fibre channel adapters and the WWPNs by using the lsmap and lsnports commands.

To enable N_Port ID Virtualization (NPIV) on the managed system, you create a pair of WWPNs for a logical partition and assign the pair directly to the physical ports of the physical fibre channel adapters. You can assign multiple logical partitions to one physical port by assigning a pair of WWPNs for each logical partition to the same physical port. When you assign a WWPN pair to a logical partition, the IVM automatically creates the following connections:

- The IVM creates a virtual fibre channel adapter on the management partition and associates it with the virtual fibre channel adapter on the logical partition.
- The IVM generates a pair of unique WWPNs and creates a virtual fibre channel adapter on the client logical partition. The IVM assigns the WWPNs to the virtual fibre channel adapter on the client logical partition, and associates the virtual fibre channel adapter on the client logical partition with the virtual fibre channel adapter on the management partition.

When you assign the WWPNs for a logical partition to a physical port, the IVM connects the virtual fibre channel adapter on the management partition to the physical port on the physical fibre channel adapter.

The IVM generates WWPNs based on the range of names available for use with the prefix in the vital product data on the managed system. This 6–digit prefix comes with the purchase of the managed system and includes 32 768 pairs of WWPNs. When you remove the connection between a logical partition and a physical port, the hypervisor deletes the WWPNs that are assigned to the virtual fibre channel adapter on the logical partition. The IVM does not reuse the deleted WWPNs when generating WWPNs for virtual fibre channel adapters in the future. If you run out of WWPNs, you must obtain an activation code that includes another prefix with 32 768 pairs of WWPNs.

To avoid configuring the physical fibre channel adapter to be a single point of failure for the connection between the logical partition and its physical storage on the storage area network (SAN), do not assign a logical partition to one physical fibre channel adapter twice. For example, do not assign a WWPN pair for a logical partition to a physical port on a physical fibre channel adapter, and then assign another WWPN pair for the same logical partition to another physical port on the same physical fibre channel adapter. Instead, assign the WWPN pairs for each logical partition to different physical fibre channel adapters.

You can add WWPN pairs for a new logical partition without assigning them to a physical port. Being able to generate WWPNs independently of a physical port assignment for a logical partition allows you to communicate these names to the SAN administrator. This ensures that the SAN administrator can configure the SAN connection appropriately such that the logical partition can connect successfully to the SAN without regard for which physical port the partition uses for the connection.

You can dynamically add or remove a WWPN pair to and from a logical partition. You can also dynamically change the physical port that is assigned to a WWPN pair.

Table 5. Dynamic logical partitioning tasks and results

Action	Result
Dynamically add a WWPN pair to a logical partition	• The IVM creates a virtual fibre channel adapter on the management partition and associates it with the virtual fibre channel adapter on the logical partition. • The IVM generates a pair of unique WWPNs and creates a virtual fibre channel adapter on the logical partition. The IVM assigns the WWPNs to the virtual fibre channel adapter on the logical partition, and associates the virtual fibre channel adapter on the logical partition with the virtual fibre channel adapter on the management partition.
Dynamically assign a WWPN pair to a physical port	The IVM connects the virtual fibre channel adapter on the management partition to the physical port on the physical fibre channel adapter.
Dynamically remove a WWPN pair from a logical partition	• The IVM removes the connection between the virtual fibre channel adapter on the management partition and the physical port on the physical fibre channel adapter. • The IVM removes the virtual fibre channel adapter from the management partition. • The IVM removes the virtual fibre channel adapter from the logical partition. The IVM deletes the WWPNs and does not reuse them.
Dynamically change the physical port assignment of a WWPN pair	The IVM changes the connection for the virtual fibre channel adapter on the management partition to the newly assigned physical port. When you change the physical port to a value of None, the IVM retains the virtual fibre channel adapter on the management partition, but removes the connection to the physical port on the physical fibre channel adapter. If you later reassign a physical port to the WWPN pair, the IVM reuses the original virtual fibre channel adapter on the management partition and connects the adapter to the newly assigned physical port.

The following table lists the Virtual I/O Server commands that you can run to view information about the fibre channel adapters.

Table 6. Virtual I/O Server commands that display information about fibre channel adapters

Virtual I/O Server command	Information displayed by command
lsmap	• Displays the virtual fibre channel adapters on the Virtual I/O Server that are connected to the physical fibre channel adapter • Displays attributes of the virtual fibre channel adapters on the client logical partitions that are associated with the virtual fibre channel adapters on the Virtual I/O Server that are connected to the physical fibre channel adapter

Table 6. Virtual I/O Server commands that display information about fibre channel adapters (continued)

Virtual I/O Server command	**Information displayed by command**
lsnports	Displays information about the physical ports on the physical fibre channel adapters that support NPIV, such as: • The name and location code of the physical port • The number of available physical ports • The total number of WWPNs that the physical port can support • Whether the switches, to which the physical fibre channel adapters are cabled, support NPIV

Virtual SCSI

Virtual SCSI allows client logical partitions to share disk storage and tape or optical devices that are assigned to the Virtual I/O Server logical partition.

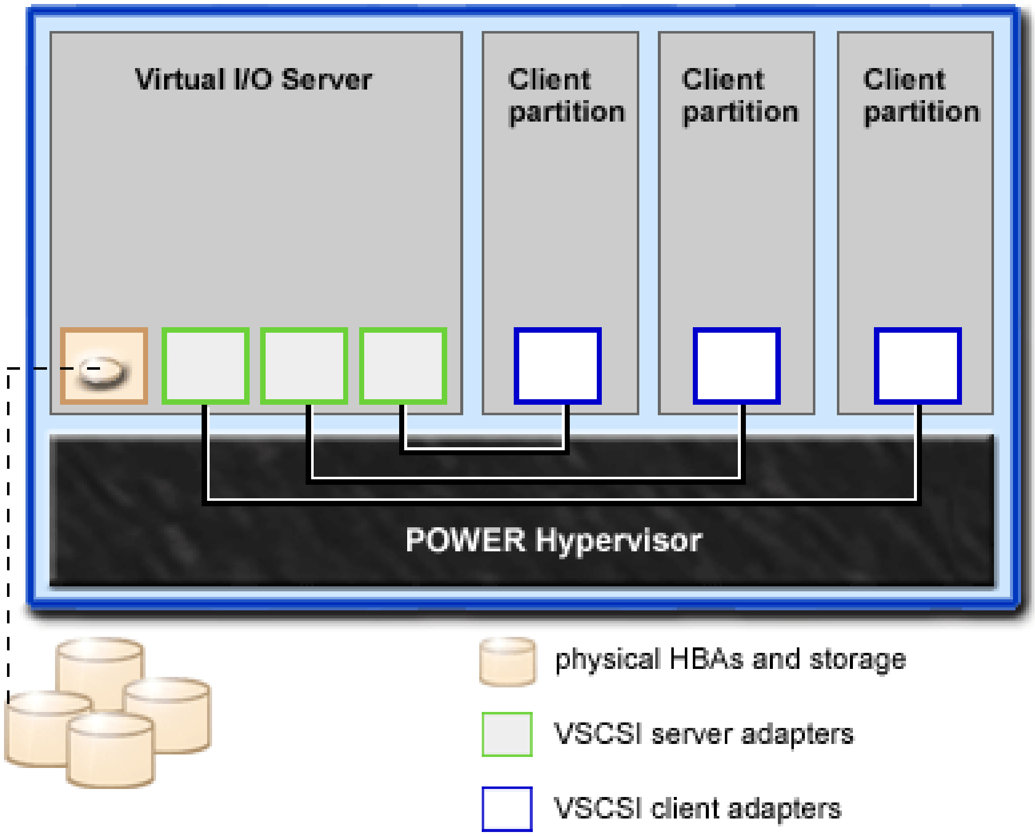
Disk, tape, or optical devices attached to physical adapters in the Virtual I/O Server logical partition can be shared by one or more client logical partitions. The Virtual I/O Server is a standard storage subsystem that provides standard SCSI-compliant LUNs. The Virtual I/O Server is capable of exporting a pool of heterogeneous physical storage as a homogeneous pool of block storage in the form of SCSI disks. The Virtual I/O Server is a localized storage subsystem. Unlike typical storage subsystems that are physically located in the SAN, the SCSI devices exported by the Virtual I/O Server are limited to the domain within the server. Therefore, although the SCSI LUNs are SCSI compliant, they might not meet the needs of all applications, particularly those that exist in a distributed environment.

The following SCSI peripheral device types are supported:

- Disk backed by logical volume
- Disk backed by physical volume
- Disk backed by file
- Optical CD-ROM, DVD-RAM, and DVD-ROM
- Optical DVD-RAM backed by file
- Tape devices

Virtual SCSI is based on a client-server relationship. The Virtual I/O Server owns the physical resources as well as the *virtual SCSI server adapter*, and acts as a server, or SCSI target device. The client logical partitions have a SCSI initiator referred to as the *virtual SCSI client adapter*, and access the virtual SCSI targets as standard SCSI LUNs. You configure the virtual adapters by using the HMC or Integrated Virtualization Manager. The configuration and provisioning of virtual disk resources is performed by using the Virtual I/O Server. Physical disks owned by the Virtual I/O Server can be either exported and assigned to a client logical partition as a whole or can be partitioned into parts, such as logical volumes or files. The logical volumes and files can then be assigned to different logical partitions. Therefore, using virtual SCSI, you can share adapters as well as disk devices. To make a physical volume, logical volume, or file available to a client logical partition requires that it be assigned to a virtual SCSI server adapter on the Virtual I/O Server. The client logical partition accesses its assigned disks through a virtual-SCSI client adapter. The virtual-SCSI client adapter recognizes standard SCSI devices and LUNs through this virtual adapter.

The following figure shows a standard virtual SCSI configuration.



Note: In order for client logical partitions to be able to access virtual devices, the Virtual I/O Server must be fully operational.

Virtual I/O Server storage subsystem overview

Learn about the Virtual I/O Server storage subsystem.

The Virtual I/O Server storage subsystem is a standard storage subsystem that provides standard SCSI-compliant LUNs. The Virtual I/O Server is a localized storage subsystem. Unlike typical storage subsystems that are physically located in the SAN, the SCSI devices exported by the Virtual I/O Server are limited to the domain within the server.

Like typical disk storage subsystems, the Virtual I/O Server has a distinct front end and back end. The front end is the interface to which client logical partitions attach to view standard SCSI-compliant LUNs. Devices on the front end are called *virtual SCSI devices*. The back end is made up of physical storage resources. These physical resources include physical disk storage, both SAN devices and internal storage devices, optical devices, tape devices, logical volumes, and files.

To create a virtual device, some physical storage must be allocated and assigned to a virtual SCSI server adapter. This process creates a virtual device instance (vtscsi*X* or vtopt*X*). The device instance can be considered a mapping device. It is not a real device, but rather a mechanism for managing the mapping of the portion of physical back-end storage to the front-end virtual SCSI device. This mapping device is instrumental in re-creating the physical-to-virtual allocations in a persistent manner when the Virtual I/O Server is restarted.

Physical storage

Learn more about physical storage, logical volumes, and the devices and configurations that are supported by the Virtual I/O Server.

Physical volumes:

Physical volumes can be exported to client partitions as virtual SCSI disks. The Virtual I/O Server is capable of taking a pool of heterogeneous physical disk storage attached to its back end and exporting this as homogeneous storage in the form of SCSI disk LUNs.

The Virtual I/O Server must be able to accurately identify a physical volume each time it boots, even if an event such as a storage area network (SAN) reconfiguration or adapter change has taken place. Physical volume attributes, such as the name, address, and location, might change after the system reboots due to SAN reconfiguration. However, the Virtual I/O Server must be able to recognize that this is the same device and update the virtual device mappings. For this reason, in order to export a physical volume as a virtual device, the physical volume must have either a unique identifier (UDID), a physical identifier (PVID), or an IEEE volume attribute.

For instructions on how to determine whether your disks have one of these identifiers, see "Identifying exportable disks" on page 102.

The following commands are used to manage physical volumes.

Table 7. Physical volume commands and their descriptions

Physical volume command	Description
lspv	Displays information about a physical volume within a volume group.
migratepv	Moves allocated physical partitions from one physical volume to one or more other physical volumes.

Logical volumes:

Understand how logical volumes can be exported to client partitions as virtual SCSI disks. A logical volume is a portion of a physical volume.

A hierarchy of structures is used to manage disk storage. Each individual disk drive or LUN, called a *physical volume*, has a name, such as **/dev/hdisk0**. Every physical volume in use either belongs to a volume group or is used directly for virtual storage. All of the physical volumes in a volume group are divided into physical partitions of the same size. The number of physical partitions in each region varies, depending on the total capacity of the disk drive.

Within each volume group, one or more logical volumes are defined. Logical volumes are groups of information located on physical volumes. Data on logical volumes appears to the user to be contiguous but can be discontiguous on the physical volume. This allows logical volumes to be resized or relocated and to have their contents replicated.

Each logical volume consists of one or more logical partitions. Each logical partition corresponds to at least one physical partition. Although the logical partitions are numbered consecutively, the underlying physical partitions are not necessarily consecutive or contiguous.

After installation, the system has one volume group (the rootvg volume group) consisting of a base set of logical volumes required to start the system.

You can use the commands described in the following table to manage logical volumes.

Table 8. Logical volume commands and their descriptions

Logical volume command	Description
chlv	Changes the characteristics of a logical volume.
cplv	Copies the contents of a logical volume to a new logical volume.
extendlv	Increases the size of a logical volume.
lslv	Displays information about the logical volume.
mklv	Creates a logical volume.
mklvcopy	Creates a copy of a logical volume.
rmlv	Removes logical volumes from a volume group.
rmlvcopy	Removes a copy of a logical volume.

Creating one or more distinct volume groups rather than using logical volumes that are created in the rootvg volume group allows you to install any newer versions of the Virtual I/O Server while maintaining client data by exporting and importing the volume groups created for virtual I/O.

Notes:

- Logical volumes used as virtual disks must be less than 1 TB (where TB equals 1 099 511 627 776 bytes) in size.
- For best performance, avoid using logical volumes (on the Virtual I/O Server) as virtual disks that are mirrored or striped across multiple physical volumes.

Volume groups:

Find information about volume groups.

A volume group is a type of storage pool that contains one or more physical volumes of varying sizes and types. A physical volume can belong to only one volume group per system. There can be up to 4096 active volume groups on the Virtual I/O Server.

When a physical volume is assigned to a volume group, the physical blocks of storage media on it are organized into physical partitions of a size determined by the system when you create the volume group. For more information, see “Physical partitions” on page 15.

When you install the Virtual I/O Server, the root volume group called rootvg is automatically created that contains the base set of logical volumes required to start the system logical partition. The rootvg includes paging space, the journal log, boot data, and dump storage, each in its own separate logical volume. The rootvg has attributes that differ from user-defined volume groups. For example, the rootvg cannot be imported or exported. When using a command or procedure on the rootvg, you must be familiar with its unique characteristics.

Table 9. Frequently used volume group commands and their descriptions

Command	Description
activatevg	Activates a volume group
chvg	Changes the attributes of a volume group
deactivatevg	Deactivates a volume group
exportvg	Exports the definition of a volume group
extendvg	Adds a physical volume to a volume group

Table 9. Frequently used volume group commands and their descriptions (continued)

Command	**Description**
importvg	Imports a new volume group definition
lsvg	Displays information about a volume group
mkvg	Creates a volume group
reducevg	Removes a physical volume from a volume group
syncvg	Synchronizes logical volume copies that are not current

Small systems might require only one volume group to contain all of the physical volumes (beyond the rootvg volume group). You can create separate volume groups to make maintenance easier because groups other than the one being serviced can remain active. Because the rootvg must always be online, it contains only the minimum number of physical volumes necessary for system operation. It is recommended that the rootvg not be used for client data.

You can move data from one physical volume to other physical volumes in the same volume group by using the migratepv command. This command allows you to free a physical volume so it can be removed from the volume group. For example, you could move data from a physical volume that is to be replaced.

Physical partitions:

This topic contains information about physical partitions.

When you add a physical volume to a volume group, the physical volume is partitioned into contiguous, equal-sized units of space called *physical partitions*. A physical partition is the smallest unit of storage space allocation and is a contiguous space on a physical volume.

Physical volumes inherit the volume group's physical partition size.

Logical partitions:

This topic contains information logical storage partitions.

When you create a logical volume, you specify its size in megabytes or gigabytes. The system allocates the number of logical partitions that are required to create a logical volume of at least the specified size. A logical partition is one or two physical partitions, depending on whether the logical volume is defined with mirroring enabled. If mirroring is disabled, there is only one copy of the logical volume (the default). In this case, there is a direct mapping of one logical partition to one physical partition. Each instance, including the first, is called a copy.

Quorums:

Find information about quorums.

A quorum exists when a majority of Volume Group Descriptor Areas and Volume Group Status Areas (VGDA/VGSA) and their disks are active. A quorum ensures data integrity of the VGDA/VGSA in the event of a disk failure. Each physical disk in a volume group has at least one VGDA/VGSA. When a volume group is created onto a single disk, the volume group initially has two VGDA/VGSA on the disk. If a volume group consists of two disks, one disk still has two VGDA/VGSA, but the other disk has one VGDA/VGSA. When the volume group is made up of three or more disks, each disk is allocated just one VGDA/VGSA.

A quorum is lost when enough disks and their VGDA/VGSA are unreachable so that a 51% majority of VGDA/VGSA no longer exists.

When a quorum is lost, the volume group deactivates itself so that the disks are no longer accessible by the logical volume manager. This prevents further disk I/O to that volume group so that data is not lost or assumed to be written when physical problems occur. As a result of the deactivation, the user is notified in the error log that a hardware error has occurred and service must be performed.

A volume group that has been deactivated because its quorum has been lost can be reactivated by using the activatevg -f command.

Virtual media repository:

The virtual media repository provides a single container to store and manage file-backed virtual optical media files. Media stored in the repository can be loaded into file-backed virtual optical devices for exporting to client partitions.

Only one repository can be created within a Virtual I/O Server.

The virtual media repository is available with Virtual I/O Server version 1.5 or later.

The virtual media repository is created and managed using the following commands.

Table 10. Virtual media repository commands and their descriptions

Command	Description
chrep	Changes the characteristics of the virtual media repository
chvopt	Changes the characteristics of a virtual optical media
loadopt	Loads file-backed virtual optical media into a file-backed virtual optical device
lsrep	Displays information about the virtual media repository
lsvopt	Displays information about file-backed virtual optical devices
mkrep	Creates the virtual media repository
mkvdev	Creates file-backed virtual optical devices
mkvopt	Creates file-backed virtual optical media
rmrep	Removes the virtual media repository
rmvopt	Removes file-backed virtual optical media
unloadopt	Unloads file-backed virtual optical media from a file-backed virtual optical device

Storage pools:

Learn about logical volume storage pools and file storage pools.

In Virtual I/O Server version 1.5 and later, you can create the following types of storage pools:

- Logical volume storage pools (LVPOOL)
- File storage pools (FBPOOL)

Like volume groups, logical volume storage pools are collections of one or more physical volumes. The physical volumes that comprise a logical volume storage pool can be of varying sizes and types. File storage pools are created within a parent logical volume storage pool and contain a logical volume containing a file system with files.

Logical volume storage pools store logical volume backing devices, file-backed storage pools, and the virtual media repository. File storage pools store file-backing devices.

Using storage pools, you are not required to have extensive knowledge of how to manage volume groups and logical volumes to create and assign logical storage to a client logical partition. Devices created using a storage pool are not limited to the size of the individual physical volumes.

Storage pools are created and managed using the following commands.

Table 11. Storage pool commands and their descriptions

Command	Description
chsp	Changes the characteristics of a storage pool
chbdsp	Changes the characteristics of a backing device within a storage pool
lssp	Displays information about a storage pool
mkbdsp	Assigns storage from a storage pool to be a backing device for a virtual SCSI adapter
mksp	Creates a storage pool
rmbdsp	Disassociates a backing device from its virtual SCSI adapter and removes it from the system
rmsp	Removes a file storage pool

Each Virtual I/O Server logical partition has a single default storage pool that can be modified only by the prime administrator. If the default storage pool is not modified by the prime administrator, rootvg, which is a logical volume pool, is used as the default storage pool.

Do not create client storage in rootvg. Creating one or more distinct logical volume storage pools rather than using the rootvg volume group allows you to install any newer versions of the Virtual I/O Server while maintaining client data by exporting and importing the volume groups created for virtual I/O.

Unless explicitly specified otherwise, the storage pool commands will operate on the default storage pool. This situation can be useful on systems that contain most or all of its backing devices in a single storage pool.

Note: Storage pools cannot be used when assigning whole physical volumes as backing devices.

Optical devices:

Optical devices can be exported by the Virtual I/O Server. This topic gives information about what types of optical devices are supported.

The Virtual I/O Server supports exporting optical SCSI devices. These are referred to as a *virtual SCSI optical devices*. Virtual optical devices can be backed by DVD drives or files. Depending on the backing device, the Virtual I/O Server will export a virtual optical device with one of following profiles:

- DVD-ROM
- DVD-RAM

Virtual optical devices that are backed by physical optical devices can be assigned to only one client logical partition at a time. In order to use the device on a different client logical partition, it must first be removed from its current logical partition and reassigned to the logical partition that will use the device.

Tape:

Tape devices can be exported by the Virtual I/O Server. This topic gives information about what types of tape devices are supported.

The Virtual I/O Server supports exporting physical tape devices to client logical partitions. These are referred to as *virtual SCSI tape devices*. Virtual SCSI tape devices are backed up by physical tape devices.

Virtual SCSI tape devices are assigned to only one client logical partition at any given time. To use the device on a different client logical partition, it must first be removed from its current logical partition and reassigned to the logical partition that will use the device.

Restriction:

- The physical tape device must be a SAS attached tape device.
- The Virtual I/O Server does not support function to move media, even if the backup device does support it.
- It is recommended that you assign the tape device to its own Virtual I/O Server adapter because as tape devices often send large amounts of data, which might affect the performance of any other device on the adapter.

Virtual storage

Disks, tapes, and optical devices are supported as virtual SCSI devices. This topic describes how those devices function in a virtualized environment and provides information on what devices are supported.

The Virtual I/O Server might virtualize, or export, disks, tapes, and optical devices, such as CD-ROM drives and DVD drives, as virtual devices. For a list of supported disks and optical devices, see the datasheet available on the Virtual I/O Server Support for UNIX® servers and Midrange servers Web site. For information about configuring virtual SCSI devices, see "Creating the virtual target device on the Virtual I/O Server" on page 94.

Disk:

Disk devices can be exported by the Virtual I/O Server. This topic gives information about what types of disks and configurations are supported.

The Virtual I/O Server supports exporting disk SCSI devices. These are referred to as *virtual SCSI disks*. All virtual SCSI disks must be backed by physical storage. The following types of physical storage can be used to back virtual disks:

- Virtual SCSI disk backed by a physical disk
- Virtual SCSI disk backed by a logical volume
- Virtual SCSI disk backed by a file

Regardless of whether the virtual SCSI disk is backed by a physical disk, logical volume, or a file, all standard SCSI rules apply to the device. The virtual SCSI device will behave as a standard SCSI-compliant disk device, and it can serve as a boot device or a Network Installation Management (NIM) target, for example.

Virtual SCSI Client Adapter Path Timeout

The virtual SCSI (VSCSI) Client Adapter Path Timeout feature allows the client adapter to detect whether a Virtual I/O Server is not responding to I/O requests. Use this feature only in configurations in which devices are available to a client logical partition from multiple Virtual I/O Servers. These configurations could be either configurations where Multipath I/O (MPIO) is being used or where a volume group is being mirrored by devices on multiple Virtual I/O Servers.

If no I/O requests issued to the VSCSI server adapter have been serviced within the number of seconds specified by the VSCSI path timeout value, one more attempt is made to contact the VSCSI server adapter, waiting up to 60 seconds for a response.

If, after 60 seconds, there is still no response from the server adapter, all outstanding I/O requests to that adapter are failed and an error is written to the client logical partition error log. If MPIO is being used, the MPIO Path Control Module will retry the I/O requests down another path. Otherwise, the failed requests will be returned to the applications. If the devices on this adapter are part of a mirrored volume group, those devices will be marked as *missing* and the Logical Volume Manager logs errors in the client logical partition error log. If one of the failed devices is the root volume group (rootvg) for the logical partition, and the rootvg is not available via another path or is not being mirrored on another Virtual I/O Server, the client logical partition is likely to shut down. The VSCSI client adapter attempts to reestablish communication with the Virtual I/O Server and logs a message in the system error log when it is able to do so. Mirrored volume groups must be manually resynchronized by running the varyonvg command when the missing devices are once again available.

A configurable VSCSI client adapter ODM attribute, **vscsi_path_to**, is provided. This attribute is used to both indicate if the feature is enabled and to store the value of the path timeout if the feature is enabled.

The system administrator sets the ODM attribute to 0 to disable the feature, or to the time, in seconds, to wait before checking if the path to the server adapter has failed. If the feature is enabled, a minimum setting of 30 seconds is required. If a setting between 0 and 30 seconds is entered, the value will be changed to 30 seconds upon the next adapter reconfiguration or reboot.

This feature is disabled by default, thus the default value of **vscsi_path_to** is 0. Exercise careful consideration when setting this value, keeping in mind that when the VSCSI server adapter is servicing the I/O request, the storage device the request is being sent to may be either local to the VIO Server or on a SAN.

The **vscsi_path_to** client adapter attribute can be set by using the SMIT utility or by using the **chdev -P** command. The attribute setting can also be viewed by using SMIT or the **lsattr** command. The setting will not take affect until the adapter is reconfigured or the machine is rebooted.

Optical:

Optical devices can be exported by the Virtual I/O Server. This topic gives information about what types of optical devices are supported.

The Virtual I/O Server supports exporting physical optical devices to client logical partitions. These are referred to as *virtual SCSI optical devices*. Virtual SCSI optical devices can be backed by DVD drives or files. Depending on the backing device, the Virtual I/O Server will export a virtual optical device with one of following profiles:
- DVD-ROM
- DVD-RAM

For example, file-backed virtual SCSI optical devices are exported as DVD-RAM devices. File-backed virtual SCSI optical devices can be backed by read-write or read-only files. Depending on the file permissions, the device can appear to contain a DVD-ROM or DVD-RAM disk. Read-write media files (DVD-RAM) cannot be loaded into more than one file-backed virtual SCSI optical device simultaneously. Read-only media files (DVD-ROM) can be loaded into multiple file-backed virtual SCSI optical devices simultaneously.

Virtual SCSI optical devices that are backed by physical optical devices can be assigned to only one client logical partition at any given time. To use the device on a different client logical partition, it must first be removed from its current logical partition and reassigned to the logical partition that will use the device.

Virtual SCSI optical devices will always appear as SCSI devices on the client logical partitions regardless of whether the device type exported from the Virtual I/O Server is a SCSI, IDE, USB device, or a file.

Tape:

Tape devices can be exported by the Virtual I/O Server. This topic gives information about what types of tape devices are supported.

The Virtual I/O Server supports exporting physical tape devices to client logical partitions. These are referred to as *virtual SCSI tape devices*. Virtual SCSI tape devices are backed up by physical tape devices.

Virtual SCSI tape devices are assigned to only one client logical partition at any given time. To use the device on a different client logical partition, it must first be removed from its current logical partition and reassigned to the logical partition that will use the device.

Restriction:

- The physical tape device must be a SAS attached tape device.
- The Virtual I/O Server does not support function to move media, even if the backup device does support it.
- It is recommended that you assign the tape device to its own Virtual I/O Server adapter because as tape devices often send large amounts of data, which might affect the performance of any other device on the adapter.

Device compatibility in a Virtual I/O Server environment:

Learn more about virtual-to-physical device compatibility in a Virtual I/O Server environment.

The virtual-to-physical device (p2v) compatibility described in this document refers only to the data on the device, not necessarily to the capabilities of the device. A device is p2v compatible when the data retrieved from that device is identical regardless of whether it is accessed directly through a physical attachment or virtually (for example, through the Virtual I/O Server). That is, every logical block (i.e. LBA 0 through LBA n-1) returns identical data for both physical and virtual devices. Device capacity must also be equal in order to claim p2v compliance.

Virtual disk devices exported by the Virtual I/O Server are referred to as virtual SCSI disks. A vitual SCSI disk device may be backed by an entire physical volume, a logical volume, a multi-path device, or a file.

Data replication (i.e. copy services) and device movement between physical and virtual environments are common operations in today's datacenter. These operations, involving devices in a virtualized environment, often have a dependency on p2v compliance.

Copy Services refer to various solutions that provide data replication function including data migration, flashcopy, point-in-time copy, and remote mirror and copy solutions. These capabilities are commonly used for disaster recovery, cloning, backup/restore, and more.

Device movement between physical and virtual environments refers to the ability to move a disk device between physical (i.e. direct-attached SAN) and virtual I/O (i.e. Virtual I/O Server-attached SAN) environments and use the disk without having to backup or restore the data. This capability is very useful for server consolidation.

The operations above may work if the device is p2v compatible. However, not all device combinations and data replication solutions have been tested by IBM. See claims by the Copy Services vendor for support claims for devices managed by Virtual I/O Server.

A device is p2v compatible if it meets the following criteria:

- It is an entire physical volume (ie LUN)
- Device capacity is identical in both physical and virtual environments
- The Virtual I/O Server is able to manage this physical volume using a UDID or iEEE ID. For more information, see the Determining if a physical volume is managed by UDID or iEEE topic below.

Devices managed by the following multipathing solutions within the Virtual I/O Server are expected to be UDID devices.

- All multipath I/O (MPIO) versions, including Subsystem Device Driver Path Control Module (SDDPCM), EMC PCM, and Hitachi Dynamic Link Manager (HDLM) PCM
- EMC PowerPath 4.4.2.2 or later
- IBM Subsystem Device Driver (SDD) 1.6.2.3 or later
- Hitachi HDLM 5.6.1 or later

Virtual SCSI devices created with earlier versions of PowerPath, HDLM, and SDD are not managed by UDID format and are not expected to be p2v compliant. The operations mentioned above (for example, data replication or movement between Virtual I/O Server and non-Virtual I/O Server environments) are not likely to work in these cases.

Determining if a physical volume is managed by UDID or IEEE:

Determine if a physical volume is or has the capability of being managed by unit device identifier (UDID) or IEEE.

In order to determine if a physical volume is or can be managed by the UDID format, the following must be verified:

- If it is an existing Virtual I/O Server LUN, determine if its format is UDID.
- If it is a LUN to be moved to Virtual I/O Server, first verify that the Virtual I/O Server is prepared to see that LUN as a UDID LUN by checking it at the source host.

  Note: Moving a physical disk to a Virtual I/O Server that is not capable of managing the device using UDID may result in data loss. In this case, backup the data prior to allocating the LUN to the Virtual I/O Server.

To determine if a device has a UDID, complete the following steps:

Note: These instructions are for Virtual I/O Server. For AIX, omit step 1 and follow the same instructions.

1. Type `oem_setup_env`.
2. Type `odmget -qattribute=unique_id CuAt`. The disks that have a UDID are listed. Output similar to the following is displayed:

   ```
   CuAt:
    name = "hdisk1"
    attribute = "unique_id"
    value = "2708ECVBZ1SC10IC35L146UCDY10-003IBMscsi"
    type = "R"
    generic = ""
    rep = "nl"
    nls_index = 79

   CuAt:
    name = "hdisk2"
    attribute = "unique_id"
    value = "210800038FB50AST373453LC03IBMscsi"
    type = "R"
    generic = ""
    rep = "nl"
   nls_index = 79
   ```
3. To determine whether a device has an IEEE volume attribute identifier, run the following command: `lsattr -l hdiskX`. Disks with an IEEE volume attribute identifier have a value in the *ieee_volname* field. Output similar to the following is displayed:

```
...
cache_method   fast_write                      Write Caching method
ieee_volname   600A0B800012DD0D00000AB441ED6AC IEEE Unique volume name
lun_id         0x001a000000000000              Logical Unit Number
  ...
```

If the *ieee_volname* field does not appear, then the device does not have an IEEE volume attribute identifier.

Note: DS4K and FAStT storage that are using the Redundant Disk Array Controller (RDAC) driver for multipathing are managed using an IEEE ID.

Mapping devices

Mapping devices are used to facilitate the mapping of physical resources to a virtual device.

Virtual networking

Learn about virtual Ethernet, Host Ethernet Adapter (or Integrated Virtual Ethernet), Internet Protocol version 6 (IPv6), Link Aggregation (or EtherChannel), Shared Ethernet Adapter, Shared Ethernet Adapter failover, and VLAN.

Virtual Ethernet technology facilitates IP-based communication between logical partitions on the same system using virtual local area network (VLAN)-capable software switch systems. Using Shared Ethernet Adapter technology, logical partitions can communicate with other systems outside the hardware unit without assigning physical Ethernet slots to the logical partitions.

Host Ethernet Adapter

A *Host Ethernet Adapter (HEA)* is a physical Ethernet adapter that is integrated directly into the GX+ bus on a managed system. HEAs offer high throughput, low latency, and virtualization support for Ethernet connections. HEAs are also known as Integrated Virtual Ethernet adapters (IVE adapters).

Unlike most other types of I/O devices, you can never assign the HEA itself to a logical partition. Instead, multiple logical partitions can connect directly to the HEA and use the HEA resources. This allows these logical partitions to access external networks through the HEA without having to go through an Ethernet bridge on another logical partition.

To connect a logical partition to an HEA, you must create a logical Host Ethernet Adapter (LHEA) for the logical partition. A *logical Host Ethernet Adapter (LHEA)* is a representation of a physical HEA on a logical partition. An LHEA appears to the operating system as if it were a physical Ethernet adapter, just as a virtual Ethernet adapter appears as if it were a physical Ethernet adapter. When you create an LHEA for a logical partition, you specify the resources that the logical partition can use on the actual physical HEA. Each logical partition can have one LHEA for each physical HEA on the managed system. Each LHEA can have one or more logical ports, and each logical port can connect to a physical port on the HEA.

You can create an LHEA for a logical partition using either of the following methods:

- You can add the LHEA to a partition profile, shut down the logical partition, and reactivate the logical partition using the partition profile with the LHEA.
- You can add the LHEA to a running logical partition using dynamic logical partitioning. (This method can be used for Linux logical partitions only if you install Red Hat Enterprise Linux version 5.1, Red Hat Enterprise Linux version 4.6, or a later version of Red Hat Enterprise Linux on the logical partition.)

When you activate a logical partition, the LHEAs in the partition profile are considered to be required resources. If the physical HEA resources required by the LHEAs are not available, then the logical partition cannot be activated. However, when the logical partition is active, you can remove any LHEAs you want from the logical partition. For every active LHEA that you assign to an IBM i logical partition, IBM i requires 40 MB of memory.

After you create an LHEA for a logical partition, a network device is created in the logical partition. This network device is named `entX` on AIX logical partitions, `CMNXX` on IBM i logical partitions, and `ethX` on Linux logical partitions, where `X` represents sequentially assigned numbers. The user can then set up TCP/IP configuration similar to a physical Ethernet device to communicate with other logical partitions.

You can configure a logical partition so that it is the only logical partition that can access a physical port of an HEA by specifying *promiscuous mode* for an LHEA that is assigned to the logical partition. When an LHEA is in promiscuous mode, no other logical partitions can access the logical ports of the physical port that is associated with the LHEA that is in promiscuous mode. You might want to configure a logical partition to promiscuous mode in the following situations:

- If you want to connect more than 16 logical partitions to each other and to an external network through a physical port on an HEA, you can create a logical port on a Virtual I/O Server logical partition and configure an Ethernet bridge between the logical port and a virtual Ethernet adapter on a virtual LAN. This allows all logical partitions with virtual Ethernet adapters on the virtual LAN to communicate with the physical port through the Ethernet bridge. If you configure an Ethernet bridge between a logical port and a virtual Ethernet adapter, the physical port that is connected to the logical port must have the following properties:
  - The physical port must be configured so that the Virtual I/O Server logical partition is the promiscuous mode logical partition for the physical port.
  - The physical port can have only one logical port.
- You want the logical partition to have dedicated access to a physical port.
- You want to use tools such as `tcpdump` or `iptrace`.

A logical port can communicate with all other logical ports that are connected to the same physical port on the HEA. The physical port and its associated logical ports form a logical Ethernet network. Broadcast and multicast packets are distributed on this logical network as though it was a physical Ethernet network. You can connect up to 16 logical ports to a physical port using this logical network. By extension, you can connect up to 16 logical partitions to each other and to an external network through this logical network. The actual number of logical ports that you can connect to a physical port depends upon the Multi-Core Scaling value of the physical port group and the number of logical ports that have been created for other physical ports within the physical port group. By default, the Multi-Core Scaling value of each physical port group is set to 4, which allows 4 logical ports to be connected to the physical ports in the physical port group. To allow up to 16 logical ports to be connected to the physical ports in the physical port group, you must change the Multi-Core Scaling value of the physical port group to 1 and restart the managed system.

You can set each logical port to restrict or allow packets that are tagged for specific VLANs. You can set a logical port to accept packets with any VLAN ID, or you can set a logical port to accept only the VLAN IDs that you specify. You can specify up to 20 individual VLAN IDs for each logical port.

The physical ports on an HEA are always configured on the managed system level. If you use an HMC to manage a system, you must use the HMC to configure the physical ports on any HEAs belonging to the managed system. Also, the physical port configuration applies to all logical partitions that use the physical port. (Some properties might require setup in the operating system as well. For example, the maximum packet size for a physical port on the HEA must be set on the managed system level using the HMC. However, you must also set the maximum packet size for each logical port within the operating system.) By contrast, if a system is unpartitioned and is not managed by an HMC, you can configure the physical ports on an HEA within the operating system just as if the physical ports were ports on a regular physical Ethernet adapter.

HEA hardware does not support half duplex mode.

You can change the properties of a logical port on an LHEA by using dynamic logical partitioning to remove the logical port from the logical partition and add the logical port back to the logical partition using the changed properties. If the operating system of the logical partition does not support dynamic logical partitioning for LHEAs, and you want to change any logical port property other than the VLANs on which the logical port participates, you must set a partition profile for the logical partition so that the partition profile contains the desired logical port properties, shut down the logical partition, and activate the logical partition using the new or changed partition profile. If the operating system of the logical partition does not support dynamic logical partitioning for LHEAs, and you want to change the VLANs on which the logical port participates, you must remove the logical port from a partition profile belonging to the logical partition, shut down and activate the logical partition using the changed partition profile, add the logical port back to the partition profile using the changed VLAN configuration, and shut down and activate the logical partition again using the changed partition profile.

Internet Protocol version 6

Internet Protocol version 6 (IPv6) is the next generation of Internet Protocol and is gradually replacing the current Internet standard, Internet Protocol version 4 (IPv4). The key IPv6 enhancement is the expansion of the IP address space from 32 bits to 128 bits, providing virtually unlimited, unique IP addresses.

IPv6 provides several advantages over IPv4, including expanded routing and addressing, routing simplification, header format simplification, improved traffic control, autoconfiguration, and security.

For more information about IPv6, see the following resources:
- AIX: Internet Protocol (IP) version 6
- IBM i: Internet Protocol version 6

Link Aggregation or EtherChannel devices

A Link Aggregation, or EtherChannel, device is a network port-aggregation technology that allows several Ethernet adapters to be aggregated. The adapters can then act as a single Ethernet device. Link Aggregation helps provide more throughput over a single IP address than would be possible with a single Ethernet adapter.

For example, `ent0` and `ent1` can be aggregated to `ent3`. The system considers these aggregated adapters as one adapter, and all adapters in the Link Aggregation device are given the same hardware address, so they are treated by remote systems as if they are one adapter.

Link Aggregation can help provide more redundancy because individual links might fail, and the Link Aggregation device will fail over to another adapter in the device to maintain connectivity. For example, in the previous example, if `ent0` fails, the packets are automatically sent on the next available adapter, `ent1`, without disruption to existing user connections. `ent0` automatically returns to service on the Link Aggregation device when it recovers.

You can configure a Shared Ethernet Adapter to use a Link Aggregation, or EtherChannel, device as the physical adapter.

Virtual Ethernet adapters

Virtual Ethernet adapters allow client logical partitions to send and receive network traffic without having a physical Ethernet adapter.

Virtual Ethernet adapters allow logical partitions within the same system to communicate without having to use physical Ethernet adapters. Within the system, virtual Ethernet adapters are connected to an IEEE 802.1q virtual Ethernet switch. Using this switch function, logical partitions can communicate with each other by using virtual Ethernet adapters and assigning VIDs. With VIDs, virtual Ethernet adapters can share a common logical network. The system transmits packets by copying the packet directly from the memory of the sender logical partition to the receive buffers of the receiver logical partition without any intermediate buffering of the packet.

Virtual Ethernet adapters can be used without using the Virtual I/O Server, but the logical partitions will not be able to communicate with external systems. However, in this situation, you can use another device, called a Host Ethernet Adapter (or Integrated Virtual Ethernet), to facilitate communication between logical partitions on the system and external networks.

You can create virtual Ethernet adapters using the Hardware Management Console (HMC) and configure them using the Virtual I/O Server command-line interface. You can also use the Integrated Virtualization Manager to create and manage virtual Ethernet adapters.

Consider using virtual Ethernet on the Virtual I/O Server in the following situations:

- When the capacity or the bandwidth requirement of the individual logical partition is inconsistent with, or is less than, the total bandwidth of a physical Ethernet adapter. Logical partitions that use the full bandwidth or capacity of a physical Ethernet adapter should use dedicated Ethernet adapters.
- When you need an Ethernet connection, but there is no slot available in which to install a dedicated adapter.

Virtual local area networks

Virtual local area networks (VLAN) allows the physical network to be logically segmented.

VLAN is a method to logically segment a physical network so that layer 2 connectivity is restricted to members that belong to the same VLAN. This separation is achieved by tagging Ethernet packets with their VLAN membership information and then restricting delivery to members of that VLAN. VLAN is described by the IEEE 802.1Q standard.

The VLAN tag information is referred to as VLAN ID (VID). Ports on a switch are configured as being members of a VLAN designated by the VID for that port. The default VID for a port is referred to as the Port VID (PVID). The VID can be added to an Ethernet packet either by a VLAN-aware host, or by the switch in the case of VLAN-unaware hosts. Ports on an Ethernet switch must therefore be configured with information indicating whether the host connected is VLAN-aware.

For VLAN-unaware hosts, a port is set up as untagged and the switch will tag all packets entering through that port with the Port VLAN ID (PVID). It will also untag all packets exiting that port before delivery to the VLAN unaware host. A port used to connect VLAN-unaware hosts is called an *untagged port*, and it can be a member of only a single VLAN identified by its PVID. Hosts that are VLAN-aware can insert and remove their own tags and can be members of more than one VLAN. These hosts are typically attached to ports that do not remove the tags before delivering the packets to the host, but will insert the PVID tag when an untagged packet enters the port. A port will only allow packets that are untagged or tagged with the tag of one of the VLANs that the port belongs to. These VLAN rules are in addition to the regular media access control (MAC) address-based forwarding rules followed by a switch. Therefore, a packet with a broadcast or multicast destination MAC is also delivered to member ports that belong to the VLAN that is identified by the tags in the packet. This mechanism ensures the logical separation of the physical network based on membership in a VLAN.

Shared Ethernet Adapters

With Shared Ethernet Adapters on the Virtual I/O Server logical partition, virtual Ethernet adapters on client logical partitions can send and receive outside network traffic.

A Shared Ethernet Adapter is a Virtual I/O Server component that bridges a physical Ethernet adapter and one or more virtual Ethernet adapters:

- The real adapter can be a physical Ethernet adapter, a Link Aggregation or EtherChannel device, or a Logical Host Ethernet Adapter . The real adapter cannot be another Shared Ethernet Adapter or a VLAN pseudo-device.
- The virtual Ethernet adapter must be a virtual I/O Ethernet adapter. It cannot be any other type of device or adapter.

Using a Shared Ethernet Adapter, logical partitions on the virtual network can share access to the physical network and communicate with stand-alone servers and logical partitions on other systems. The Shared Ethernet Adapter eliminates the need for each client logical partition to a dedicated physical adapter to connect to the external network.

A Shared Ethernet Adapter provides access by connecting the internal VLANs with the VLANs on the external switches. Using this connection, logical partitions can share the IP subnet with stand-alone systems and other external logical partitions. The Shared Ethernet Adapter forwards outbound packets received from a virtual Ethernet adapter to the external network and forwards inbound packets to the appropriate client logical partition over the virtual Ethernet link to that logical partition. The Shared Ethernet Adapter processes packets at layer 2, so the original MAC address and VLAN tags of the packet are visible to other systems on the physical network.

The Shared Ethernet Adapter has a bandwidth apportioning feature, also known as Virtual I/O Server quality of service (QoS). QoS allows the Virtual I/O Server to give a higher priority to some types of packets. In accordance with the IEEE 801.q specification, Virtual I/O Server administrators can instruct the Shared Ethernet Adapter to inspect bridged VLAN-tagged traffic for the VLAN priority field in the VLAN header. The 3-bit VLAN priority field allows each individual packet to be prioritized with a value from 0 to 7 to distinguish more important traffic from less important traffic. More important traffic is sent preferentially and uses more Virtual I/O Server bandwidth than less important traffic.

Note: To use this feature, when the Virtual I/O Server Trunk Virtual Ethernet Adapter is configured on an HMC, the adapter must be configured with additional VLAN IDs because only the traffic on these VLAN IDs is delivered to the Virtual I/O Server with a VLAN tag. Untagged traffic is always treated as though it belonged to the default priority class that is, as if it had a priority value of 0.

Depending on the VLAN priority values found in the VLAN headers, packets are prioritized as follows.

Table 12. VLAN traffic priority values and relative importance

Priority value and importance
1 (Most important)
2
0 (Default)
3
4
5
6
7 (Least important)

The Virtual I/O Server administrator can use QoS by setting the Shared Ethernet Adapter qos_mode attribute to either strict or loose mode. The default is disabled mode. The following definitions describe these modes:

disabled mode
This is the default mode. VLAN traffic is not inspected for the priority field. An example follows:

```
chdev -dev <SEA device name> -attr qos_mode=disabled
```

strict mode
More important traffic is bridged over less important traffic. This mode provides better performance and more bandwidth to more important traffic; however, it can result in substantial delays for less important traffic. An example follows:

```
chdev -dev <SEA device name> -attr qos_mode=strict
```

loose mode
A cap is placed on each priority level so that after a number of bytes is sent for each priority level, the following level is serviced. This method ensures that all packets are eventually sent. More important traffic is given less bandwidth with this mode than with strict mode; however,

the caps in loose mode are such that more bytes are sent for the more important traffic, so it still gets more bandwidth than less important traffic. An example follows:

```
chdev -dev <SEA device name> -attr qos_mode=loose
```

Note: In either strict or loose mode, because the Shared Ethernet Adapter uses several threads to bridge traffic, it is still possible for less important traffic from one thread to be sent before more important traffic of another thread.

GARP VLAN Registration Protocol

Shared Ethernet Adapters, in Virtual I/O Server version 1.4 or later, support GARP VLAN Registration Protocol (GVRP), which is based on Generic Attribute Registration Protocol (GARP). GVRP allows for the dynamic registration of VLANs over networks, which can reduce the number of errors in the configuration of a large network. By propagating registration across the network through the transmission of Bridge Protocol Data Units (BPDUs), devices on the network have accurate knowledge of the bridged VLANs configured on the network.

When GVRP is enabled, communication travels one way, from the Shared Ethernet Adapter to the switch. The Shared Ethernet Adapter notifies the switch which VLANs can communicate with the network. The Shared Ethernet Adapter does not configure VLANs to communicate with the network based on information received from the switch. Rather, the configuration of VLANs that communicate with the network is statically determined by the virtual Ethernet adapter configuration settings.

Host Ethernet Adapter or Integrated Virtual Ethernet

With Virtual I/O Server version 1.4, you can assign a logical host Ethernet port, of a logical host Ethernet adapter (LHEA), which is sometimes referred to as Integrated Virtual Ethernet, as the real adapter of a Shared Ethernet Adapter. The logical host Ethernet port is associated with a physical port on the Host Ethernet Adapter. The Shared Ethernet Adapter uses the standard device driver interfaces provided by the Virtual I/O Server to communicate with the Host Ethernet Adapter.

To use a Shared Ethernet Adapter with a Host Ethernet Adapter, the following requirements must be met:

- The logical host Ethernet port must be the only port assigned to the physical port on the Host Ethernet Adapter. No other ports of the LHEA can be assigned to the physical port on the Host Ethernet Adapter.
- The LHEA on the Virtual I/O Server logical partition must be set to `promiscuous` mode. (In an Integrated Virtualization Manager environment, the mode is set to `promiscuous` by default.) *Promiscuous mode* allows the LHEA (on the Virtual I/O Server) to receive all unicast, multicast, and broadcast network traffic from the physical network.

Recommendations

Consider using Shared Ethernet Adapters on the Virtual I/O Server in the following situations:

- When the capacity or the bandwidth requirement of the individual logical partition is inconsistent or is less than the total bandwidth of a physical Ethernet adapter. Logical partitions that use the full bandwidth or capacity of a physical Ethernet adapter should use dedicated Ethernet adapters.
- If you plan to migrate a client logical partition from one system to another.

Consider assigning a Shared Ethernet Adapter to a Logical Host Ethernet port when the number of Ethernet adapters that you need is more than the number of ports available on the LHEA, or you anticipate that your needs will grow beyond that number. If the number of Ethernet adapters that you need is fewer than or equal to the number of ports available on the LHEA, and you do not anticipate needing more ports in the future, you can use the ports of the LHEA for network connectivity rather than the Shared Ethernet Adapter.

Shared memory

Shared memory is physical memory that is assigned to the shared memory pool and shared among multiple logical partitions. The *shared memory pool* is a defined collection of physical memory blocks that are managed as a single memory pool by the hypervisor. Logical partitions that you configure to use shared memory (hereafter referred to as *shared memory partitions*) share the memory in the pool with other shared memory partitions.

For example, you create a shared memory pool with 16 GB of physical memory. You then create three logical partitions, configure them to use shared memory, and activate the shared memory partitions. Each shared memory partition can use the 16 GB that are in the shared memory pool.

The hypervisor determines the amount of memory allocated from the shared memory pool to each shared memory partition based on the workload and memory configuration of each shared memory partition. When allocating the physical memory to the shared memory partitions, the hypervisor ensures that each shared memory partition can access only the memory allocated to the shared memory partition at any given time. A shared memory partition cannot access the physical memory allocated to another shared memory partition.

The amount of memory that you assign to the shared memory partitions can be greater than the amount of memory in the shared memory pool. For example, you can assign 12 GB to shared memory partition 1, 8 GB to shared memory partition 2, and 4 GB to shared memory partition 3. Together, the shared memory partitions use 24 GB of memory, but the shared memory pool has only 16 GB of memory. In this situation, the memory configuration is considered overcommitted.

Overcommitted memory configurations are possible because the hypervisor virtualizes and manages all of the memory for the shared memory partitions in the shared memory pool as follows:

1. When shared memory partitions are not actively using their memory pages, the hypervisor allocates those unused memory pages to shared memory partitions that currently need them. When the sum of the physical memory currently used by the shared memory partitions is less than or equal to the amount of memory in the shared memory pool, the memory configuration is *logically overcommitted*. In a logically overcommitted memory configuration, the shared memory pool has enough physical memory to contain the memory used by all shared memory partitions at one point in time. The hypervisor does not need to store any data in auxiliary storage.
2. When a shared memory partition requires more memory than the hypervisor can provide to it by allocating unused portions of the shared memory pool, the hypervisor stores some of the memory that belongs to a shared memory partition in the shared memory pool and stores the remainder of the memory that belongs to the shared memory partition in auxiliary storage. When the sum of the physical memory currently used by the shared memory partitions is greater than the amount of memory in the shared memory pool, the memory configuration is *physically overcommitted*. In a physically overcommitted memory configuration, the shared memory pool does not have enough physical memory to contain the memory used by all the shared memory partitions at one point in time. The hypervisor stores the difference in auxiliary storage. When the operating system attempts to access the data, the hypervisor might need to retrieve the data from auxiliary storage before the operating system can access it.

Because the memory that you assign to a shared memory partition might not always reside in the shared memory pool, the memory that you assign to a shared memory partition is *logical memory*. Logical memory is the address space, assigned to a logical partition, that the operating system perceives as its main storage. For a shared memory partition, a subset of the logical memory is backed up by physical main storage (or physical memory from the shared memory pool) and the contents of the remaining logical memory are kept in auxiliary storage.

A Virtual I/O Server logical partition provides access to the auxiliary storage, or paging space devices, required for shared memory partitions in an overcommitted memory configuration. A *paging space device* is a physical or logical device that is used by a Virtual I/O Server to provide the paging space for a shared memory partition. The *paging space* is an area of nonvolatile storage used to hold portions of a shared memory partition's logical memory that do not reside in the shared memory pool. When the operating system that runs in a shared memory partition attempts to access data, and the data is located in the paging space device that is assigned to the shared memory partition, the hypervisor sends a request to a Virtual I/O Server to retrieve the data and write it to the shared memory pool so that the operating system can access it.

On systems that are managed by a Hardware Management Console (HMC), you can assign up to two Virtual I/O Server (VIOS) logical partitions to the shared memory pool at a time (hereafter referred to as *paging VIOS partitions*). When you assign two paging VIOS partitions to the shared memory pool, you can configure the paging space devices such that both paging VIOS partitions have access to the same paging space devices. When one paging VIOS partition becomes unavailable, the hypervisor sends a request to the other paging VIOS partition to retrieve the data on the paging space device.

You cannot configure paging VIOS partitions to use shared memory. Paging VIOS partitions do not use the memory in the shared memory pool. You assign paging VIOS partitions to the shared memory pool so that they can provide access to the paging space devices for the shared memory partitions that are assigned to the shared memory pool.

Driven by workload demands from the shared memory partitions, the hypervisor manages overcommitted memory configurations by continually performing the following tasks:

- Allocating portions of physical memory from the shared memory pool to the shared memory partitions as needed
- Requesting a paging VIOS partition to read and write data between the shared memory pool and the paging space devices as needed

The ability to share memory among multiple logical partitions is known as the PowerVM Active Memory Sharing technology. The PowerVM Active Memory Sharing technology is available with the PowerVM Enterprise Edition for which you must obtain and enter a PowerVM Editions activation code.

Related reference

"Configuration requirements for shared memory" on page 68
Review the requirements for the system, Virtual I/O Server (VIOS), logical partitions, and paging space devices so that you can successfully configure shared memory.

Related information

Paging space device

Paging VIOS partition

A Virtual I/O Server (VIOS) logical partition that is assigned to the shared memory pool (hereafter referred to as a *paging VIOS partition*) provides access to the paging space devices for the logical partitions that are assigned to the shared memory pool (hereafter referred to as *shared memory partitions*).

When the operating system that runs in a shared memory partition attempts to access data, and the data is located in the paging space device that is assigned to the shared memory partition, the hypervisor sends a request to a paging VIOS partition to retrieve the data and write it to the shared memory pool so that the operating system can access it.

A paging VIOS partition is not a shared memory partition and does not use the memory in the shared memory pool. A paging VIOS partition provides access to the paging space devices for the shared memory partitions.

Integrated Virtualization Manager

On systems that are managed by the Integrated Virtualization Manager, the management partition is the paging VIOS partition for the shared memory partitions that are assigned to the shared memory pool.

When you create the shared memory pool, you assign a paging storage pool to the shared memory pool. The paging storage pool provides the paging space devices for the shared memory partitions that are assigned to the shared memory pool.

HMC

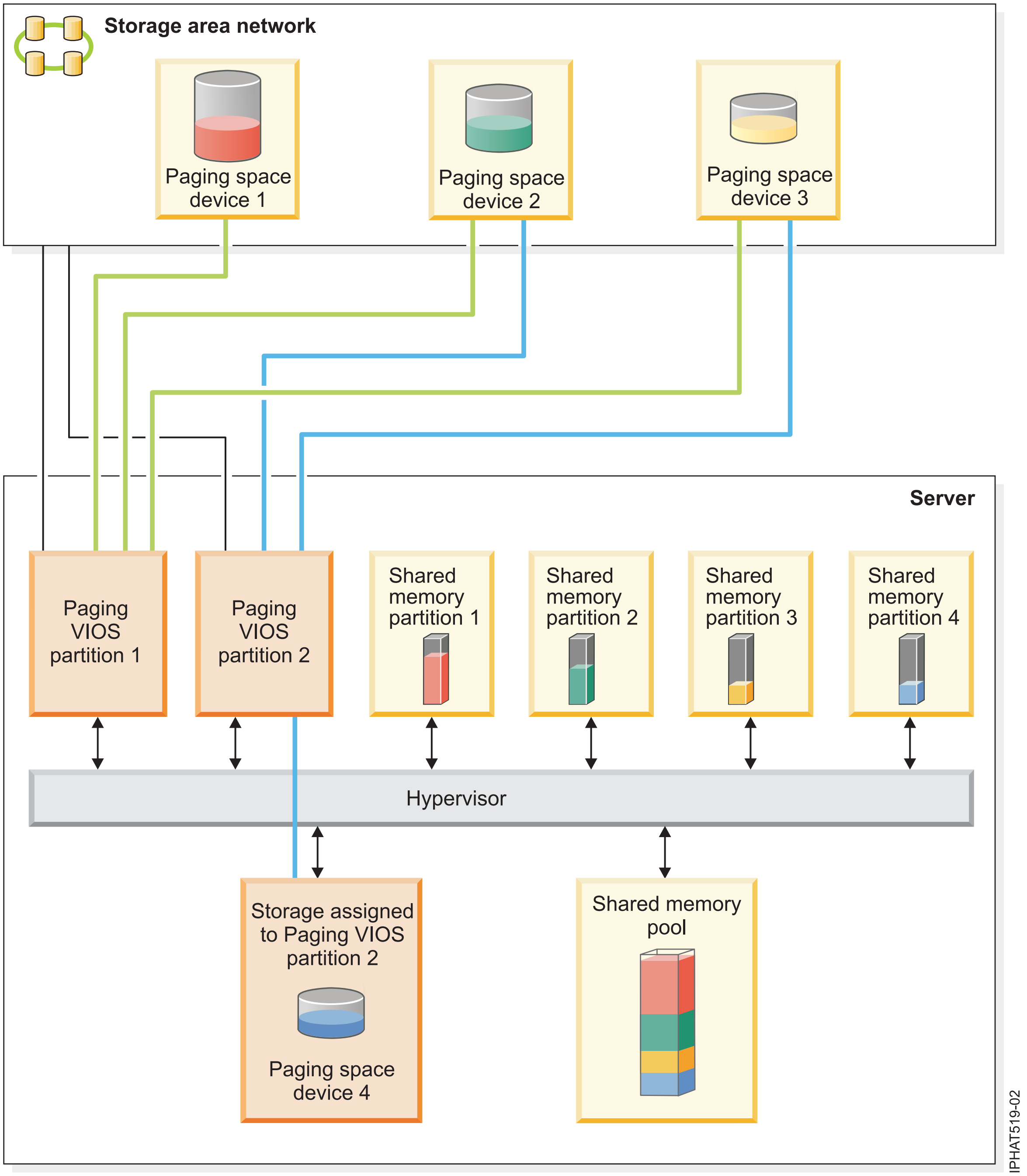
On systems that are managed by a Hardware Management Console (HMC), you can assign one or two paging VIOS partitions to the shared memory pool. When you assign a single paging VIOS partition to the shared memory pool, the paging VIOS partition provides access to all of the paging space devices for the shared memory partitions. The paging space devices can be located in physical storage in the server or on a storage area network (SAN). When you assign two paging VIOS partitions to the shared memory pool, you can configure each paging VIOS partition to access paging space devices in one of the following ways:

- You can configure each paging VIOS partition to access independent paging space devices. Paging space devices that are accessed by only one paging VIOS partition, or independent paging space devices, can be located in physical storage in the server or on a SAN.
- You can configure both paging VIOS partitions to access the same, or common, paging space devices. In this configuration, the paging VIOS partitions provide redundant access to paging space devices. When one paging VIOS partition becomes unavailable, the hypervisor sends a request to the other paging VIOS partition to retrieve the data on the paging space device. Common paging space devices must be located on a SAN to enable symmetrical access from both paging VIOS partitions.
- You can configure each paging VIOS partition to access some independent paging space devices and some common paging space devices.

If you configure the shared memory pool with two paging VIOS partitions, you can configure a shared memory partition to use either a single paging VIOS partition or redundant paging VIOS partitions. When you configure a shared memory partition to use redundant paging VIOS partitions, you assign a primary paging VIOS partition and a secondary paging VIOS partition to the shared memory partition. The hypervisor uses the primary paging VIOS partition to access the shared memory partition's paging space device. At this point, the primary paging VIOS partition is the current paging VIOS partition for the shared memory partition. The current paging VIOS partition is the paging VIOS partition that the hypervisor uses at any point in time to access data in the paging space device that is assigned to the shared memory partition. If the primary VIOS partition becomes unavailable, the hypervisor uses the secondary paging VIOS partition to access the shared memory partition's paging space device. At this point, the secondary paging VIOS partition becomes the current paging VIOS partition for the shared memory partition and continues as the current paging VIOS partition even after the primary paging VIOS partition becomes available again.

You do not need to assign the same primary and secondary paging VIOS partitions to all of the shared memory partitions. For example, you assign paging VIOS partition A and paging VIOS partition B to the shared memory pool. For one shared memory partition, you can assign paging VIOS partition A as the primary paging VIOS partition and paging VIOS partition B as the secondary paging VIOS partition. For a different shared memory partition, you can assign paging VIOS partition B as the primary paging VIOS partition and paging VIOS partition A as the secondary paging VIOS partition.

The following figure shows an example of a system with four shared memory partitions, two paging VIOS partitions, and four paging space devices.



The example shows the configuration options for paging VIOS partitions and paging space devices as described in the following table.

Table 13. Examples of paging VIOS partition configurations

Configuration option	**Example**
The paging space device that is assigned to a shared memory partition is located in physical storage in the server and is accessed by a single paging VIOS partition.	Paging space device 4 provides the paging space for Shared memory partition 4. Shared memory partition 4 is assigned to use Paging VIOS partition 2 to access Paging space device 4. Paging space device 4 is located in physical storage in the server and is assigned to Paging VIOS partition 2. Paging VIOS partition 2 is the only paging VIOS partition that can access Paging space device 4 (This relationship is shown by the blue line that connects Paging VIOS partition 2 to Paging space device 4.).
The paging space device that is assigned to a shared memory partition is located on a SAN and is accessed by a single paging VIOS partition.	Paging space device 1 provides the paging space for Shared memory partition 1. Shared memory partition 1 is assigned to use Paging VIOS partition 1 to access Paging space device 1. Paging space device 1 is connected to the SAN. Paging VIOS partition 1 is also connected to the SAN and is the only paging VIOS partition that can access Paging space device 1 (This relationship is shown by the green line that connects Paging VIOS partition 1 to Paging space device 1.).

Table 13. Examples of paging VIOS partition configurations (continued)

Configuration option	Example
The paging space device that is assigned to a shared memory partition is located on a SAN and is accessed redundantly by two paging VIOS partitions.	Paging space device 2 provides the paging space for Shared memory partition 2. Paging space device 2 is connected to the SAN. Paging VIOS partition 1 and Paging VIOS partition 2 are also connected to the SAN and can both access Paging space device 2. (These relationships are shown by the green line that connects Paging VIOS partition 1 to Paging space device 2 and the blue line that connects Paging VIOS partition 2 to Paging space device 2.) Shared memory partition 2 is assigned to use redundant paging VIOS partitions to access Paging space device 2. Paging VIOS partition 1 is configured as the primary paging VIOS partition and Paging VIOS partition 2 is configured as the secondary paging VIOS partition. Similarly, Paging space device 3 provides the paging space device for Shared memory partition 3. Paging space device 3 is connected to the SAN. Paging VIOS partition 1 and Paging VIOS partition 2 are also connected to the SAN and can both access Paging space device 3. (These relationships are shown by the green line that connects Paging VIOS partition 1 to Paging space device 3 and the blue line that connects Paging VIOS partition 2 to Paging space device 3.) Shared memory partition 3 is assigned to use redundant paging VIOS partitions to access Paging space device 3. Paging VIOS partition 2 is configured as the primary paging VIOS partition and Paging VIOS partition 1 is configured as the secondary paging VIOS partition. Because Paging VIOS partition 1 and Paging VIOS partition 2 both have access to Paging space device 2 and Paging space device 3, Paging space device 2 and Paging space device 3 are common paging space devices that are accessed redundantly by Paging VIOS partition 1 and Paging VIOS partition 2. If Paging VIOS partition 1 becomes unavailable and Shared memory partition 2 needs to access data on its paging space device, the hypervisor sends a request to Paging VIOS partition 2 to retrieve the data on Paging space device 2. Similarly, if Paging VIOS partition 2 becomes unavailable and Shared memory partition 3 needs to access the data on its paging space device, the hypervisor sends a request to Paging VIOS partition 1 to retrieve the data on Paging space device 3.

Table 13. Examples of paging VIOS partition configurations (continued)

Configuration option	**Example**
A paging VIOS partition accesses both independent and common paging space devices.	Paging space device 1 and Paging space device 4 are independent paging space devices because only one paging VIOS partition accesses each. Paging VIOS partition 1 accesses Paging space device 1, and Paging VIOS partition 2 accesses Paging space device 4. Paging space device 2 and paging space device 3 are common paging space devices because both paging VIOS partitions access each. (These relationships are shown by the green and blue lines that connect the paging VIOS partitions to the paging space devices.) Paging VIOS partition 1 accesses the independent paging space device Paging space device 1, and also accesses the common paging space devices Paging space device 2 and Paging space device 3. Paging VIOS partition 2 accesses the independent paging space device Paging space device 4 and also accesses the common paging space devices Paging space device 2 and Paging space device 3.

When a single paging VIOS partition is assigned to the shared memory pool, you must shut down the shared memory partitions before you shut down the paging VIOS partition so that the shared memory partitions are not suspended when they attempt to access their paging space devices. When two paging VIOS partitions are assigned to the shared memory pool and the shared memory partitions are configured to use redundant paging VIOS partitions, you do not need to shut down the shared memory partitions to shut down a paging VIOS partition. When one paging VIOS partition is shut down, the shared memory partitions use the other paging VIOS partition to access their paging space devices. For example, you can shut down a paging VIOS partition and install VIOS updates without shutting down the shared memory partitions.

You can configure multiple VIOS logical partitions to provide access to paging space devices. However, you can only assign up to two of those VIOS partitions to the shared memory pool at any given time.

After you configure the shared memory partitions, you can later change the redundancy configuration of the paging VIOS partitions for a shared memory partition by modifying the partition profile of the shared memory partition and restarting the shared memory partition with the modified partition profile:

- You can change which paging VIOS partitions are assigned to a shared memory partition as the primary and secondary paging VIOS partitions.
- You can change the number of paging VIOS partitions that are assigned to a shared memory partition.

Virtual I/O Server management

Learn about management tools for the Virtual I/O Server, such as the Virtual I/O Server command-line interface, and several Tivoli® products that can manage different aspects of the Virtual I/O Server.

For systems that are not managed by a Hardware Management Console (HMC), the Virtual I/O Server becomes the management partition and provides a graphical user interface, called the Integrated Virtualization Manager, to help you manage the system. For more information, see Integrated Virtualization Manager.

Virtual I/O Server command-line interface

Learn about accessing and using the Virtual I/O Server command-line interface.

The Virtual I/O Server is configured and managed through a command-line interface. In environments where no HMC is present, some Virtual I/O Server tasks can also be performed using the Integrated Virtualization Manager. All aspects of Virtual I/O Server administration can be accomplished through the command-line interface, including the following:

- Device management (physical, virtual, logical volume manager (LVM))
- Network configuration
- Software installation and update
- Security
- User management
- Maintenance tasks

In addition, in environments managed by the Integrated Virtualization Manager, you can use the Virtual I/O Server command-line interface to manage logical partitions.

The first time you log in to the Virtual I/O Server, use the **padmin** user ID, which is the prime administrator user ID. You will be prompted for a new password.

Restricted shell

Upon logging in, you will be placed into a restricted Korn shell. The restricted Korn shell works in the same way as a standard Korn shell, except that you cannot do the following:

- Change the current working directory
- Set the value of the **SHELL**, **ENV**, or **PATH** variables
- Specify the path name of the command that contains a forward slash (/)
- Redirect output of a command using any of the following characters: >, >|, <>, >>

As a result of these restrictions, you will not be able to execute commands that are not accessible to your **PATH** variables. In addition, these restrictions prevent you from sending command output directly to a file. Instead, command output can be piped to the tee command.

After you log in, you can type `help` to get information about the supported commands. For example, to get help on the errlog command, type `help errlog`.

Execution mode

The Virtual I/O Server command-line interface functions similarly to a standard command-line interface. Commands are issued with appropriate accompanying flags and parameters. For example, to list all adapters, type the following:

```
lsdev -type adapter
```

In addition, scripts can be run within the Virtual I/O Server command-line interface environment.

In addition to the Virtual I/O Server command-line interface commands, the following standard shell commands are provided.

Table 14. Standard shell commands and their functions

Command	Function
awk	Matches patterns and performs actions on them.
cat	Concatenates or displays files.
chmod	Changes file modes.
cp	Copies files.

Table 14. Standard shell commands and their functions (continued)

Command	Function
date	Displays the date and time.
grep	Searches a file for a pattern.
ls	Displays the contents of a directory
mkdir	Makes a directory.
man	Displays manual entries for the Virtual I/O Server commands.
more	Displays the contents of files one screen at a time.
rm	Removes files.
sed	Provides a stream editor.
stty	Sets, resets, and reports workstation operating parameters.
tee	Displays the output of a program and copies it to a file.
vi	Edits files with full screen display.
wc	Counts the number of lines, words, bytes, and characters in a file
who	Identifies the users currently logged in.

As each command is executed, the user log and the global command log are updated.

The user log will contain a list of each Virtual I/O Server command, including arguments, that a user has executed. One user log for each user in the system is created. This log is located in the user's home directory and can be viewed by using either the cat or the vi commands.

The global command log is made up of all the Virtual I/O Server command-line interface commands executed by all users, including arguments, the date and time the command was executed, and from which user ID it was executed. The global command log is viewable only by the **padmin** user ID, and it can be viewed by using the lsgcl command. If the global command log exceeds 1 MB, the log will be truncated to 250 KB to prevent the file system from reaching capacity.

Note: Integrated Virtualization Manager commands are audited in a separate place and are viewable either in **Application Logs**, or by running the following command from the command line:

```
lssvcevents -t console --filter severities=audit
```

Related information

Virtual I/O Server and Integrated Virtualization Manager commands

IBM Tivoli software and the Virtual I/O Server

Learn about integrating the Virtual I/O Server into your Tivoli environment for IBM Tivoli Application Dependency Discovery Manager, IBM Tivoli Monitoring, IBM Tivoli Storage Manager, IBM Tivoli Usage and Accounting Manager, IBM Tivoli Identity Manager, and IBM TotalStorage® Productivity Center.

IBM Tivoli Application Dependency Discovery Manager

IBM Tivoli Application Dependency Discovery Manager (TADDM) discovers infrastructure elements found in the typical data center, including application software, hosts and operating environments (including the Virtual I/O Server), network components (such as routers, switches, load balancers, firewalls, and storage), and network services (such as LDAP, NFS, and DNS). Based on the data it collects, TADDM automatically creates and maintains application infrastructure maps that include runtime dependencies, configuration values, and change history. With this information, you can determine the interdependences between business applications, software applications, and physical components to help you ensure and improve application availability in your environment. For example, you can do the following tasks:

- You can isolate configuration-related application problems.
- You can plan for application changes to minimize or eliminate unplanned disruptions.
- You can create a shared topological definition of applications for use by other management applications.
- You can determine the effect of a single configuration change on a business application or service.
- You can see what changes take place in the application environment and where.

TADDM includes an agent-free discovery engine, which means that the Virtual I/O Server does not require that an agent or client be installed and configured in order to be discovered by TADDM. Instead, TADDM uses discovery sensors that rely on open and secure protocols and access mechanisms to discover the data center components.

IBM Tivoli Identity Manager

With IBM Tivoli Identity Manager, you can manage identities and users across several platforms, including AIX, Windows®, Solaris, and so on. With Tivoli Identity Manager 4.7 and later, you can also include Virtual I/O Server users. Tivoli Identity Manager provides a Virtual I/O Server adapter that acts as an interface between the Virtual I/O Server and the Tivoli Identity Manager Server. The adapter might not be located on the Virtual I/O Server and the Tivoli Identity Manager Server manages access to the Virtual I/O Server by using your security system.

The adapter runs as a service, independent of whether a user is logged on to the Tivoli Identity Manager Server. The adapter acts as a trusted virtual administrator on the Virtual I/O Server, performing tasks like the following:

- Creating a user ID to authorize access to the Virtual I/O Server.
- Modifying an existing user ID to access the Virtual I/O Server.
- Removing access from a user ID. This deletes the user ID from the Virtual I/O Server.
- Suspending a user account by temporarily deactivating access to the Virtual I/O Server.
- Restoring a user account by reactivating access to the Virtual I/O Server.
- Changing a user account password on the Virtual I/O Server.
- Reconciling the user information of all current users on the Virtual I/O Server.
- Reconciling the user information of a particular user account on the Virtual I/O Server by performing a lookup.

IBM Tivoli Monitoring

Virtual I/O Server V1.3.0.1 (fix pack 8.1), includes the IBM Tivoli Monitoring System Edition for System p® agent. With Tivoli Monitoring System Edition for System p, you can monitor the health and availability of multiple IBM System p servers (including the Virtual I/O Server) from the Tivoli Enterprise Portal. Tivoli Monitoring System Edition for System p gathers data from the Virtual I/O Server, including data about physical volumes, logical volumes, storage pools, storage mappings, network mappings, real memory, processor resources, mounted file system sizes, and so on. From the Tivoli Enterprise Portal, you can view a graphical representation of the data, use predefined thresholds to alert you on key metrics, and resolve issues based on recommendations provided by the Expert Advice feature of Tivoli Monitoring.

IBM Tivoli Storage Manager

Virtual I/O Server 1.4 includes the IBM Tivoli Storage Manager client. With Tivoli Storage Manager, you can protect Virtual I/O Server data from failures and other errors by storing backup and disaster-recovery data in a hierarchy of offline storage. Tivoli Storage Manager can help protect computers running a variety of different operating environments, including the Virtual I/O Server, on a variety of different hardware, including IBM System p servers. If you configure the Tivoli Storage Manager client on the Virtual I/O Server, you can include the Virtual I/O Server in your standard backup framework.

IBM Tivoli Usage and Accounting Manager

Virtual I/O Server 1.4 includes the IBM Tivoli Usage and Accounting Manager agent on the Virtual I/O Server. Tivoli Usage and Accounting Manager helps you track, allocate, and invoice your IT costs by collecting, analyzing, and reporting on the actual resources used by entities such as cost centers, departments, and users. Tivoli Usage and Accounting Manager can gather data from multi-tiered datacenters that include Windows, AIX, Virtual I/O Server, HP/UX Sun Solaris, Linux, IBM i, and VMware.

IBM TotalStorage Productivity Center

With Virtual I/O Server 1.5.2, you can configure the IBM TotalStorage Productivity Center agents on the Virtual I/O Server. TotalStorage Productivity Center is an integrated, storage infrastructure management suite that is designed to help simplify and automate the management of storage devices, storage networks, and capacity utilization of file systems and databases. When you install and configure the TotalStorage Productivity Center agents on the Virtual I/O Server, you can use the TotalStorage Productivity Center user interface to collect and view information about the Virtual I/O Server. You can then perform the following tasks using the TotalStorage Productivity Center user interface:

1. Run a discovery job for the agents on the Virtual I/O Server.
2. Run probes, run scans, and ping jobs to collect storage information about the Virtual I/O Server.
3. Generate reports using the Fabric Manager and the Data Manager to view the storage information gathered.
4. View the storage information gathered using the topology Viewer.

Related tasks

"Configuring the IBM Tivoli agents and clients on the Virtual I/O Server" on page 109
You can configure and start the IBM Tivoli Monitoring agent, IBM Tivoli Usage and Accounting Manager, the IBM Tivoli Storage Manager client, and the IBM Tivoli TotalStorage Productivity Center agents.

Related information

IBM Tivoli Application Dependency Discovery Manager Information Center

IBM Tivoli Identity Manager

IBM Tivoli Monitoring version 6.2.1 documentation

IBM Tivoli Monitoring Virtual I/O Server Premium Agent User's Guide

IBM Tivoli Storage Manager

IBM Tivoli Usage and Accounting Manager Information Center

IBM TotalStorage Productivity Center Information Center

IBM Systems Director software

Learn about integrating the Virtual I/O Server into your IBM Systems Director environment.

IBM Systems Director is a platform-management foundation that streamlines the way you manage physical and virtual systems across a heterogeneous environment. By leveraging industry standards, IBM Systems Director supports multiple operating systems and virtualization technologies across IBM and non-IBM platforms.

Through a single user interface, IBM Systems Director provides consistent views for viewing managed systems, determining how these systems relate to one another, and identifying their statuses, thus helping to correlate technical resources with business needs. A set of common tasks included with IBM Systems Director provides many of the core capabilities that are required for basic management. These common tasks include discovery, inventory, configuration, system health, monitoring, updates, event notification and automation across managed systems.

IBM Systems Director's Web and command-line interfaces provide a consistent interface focused on these common tasks:

- Discovering, navigating and visualizing systems on the network with the detailed inventory and relationships to the other network resources
- Notifying users of problems that occur on systems and the ability to navigate to the source of the problem
- Notifying users when systems need updates, and distributing and installing updates on a schedule
- Analyzing real-time data for systems, and setting critical thresholds that notify the administrator of emerging problems
- Configuring settings of a single system, and creating a configuration plan that can apply those settings to multiple systems
- Updating installed plug-ins to add new features and function to the base capabilities
- Managing the life cycle of virtual resources

Related tasks

"Configuring the IBM Director agent" on page 115
You can configure and start the IBM Director agent on the Virtual I/O Server.

Related information

IBM Systems Director technical overview

Configuration scenarios for the Virtual I/O Server

The following scenarios show examples of networking configurations for the Virtual I/O Server logical partition and the client logical partitions. Use the following scenarios and configuration examples to understand more about the Virtual I/O Server and its components.

Scenario: Configuring a Virtual I/O Server without VLAN tagging

Use this scenario to help you become familiar with creating a network without VLAN tagging.

Situation

You are the system administrator responsible for planning and configuring the network in an environment with the Virtual I/O Server running. You want to configure a single logical subnet on the system that communicates with the switch.

Objective

The objective of this scenario is to configure the network where only Port Virtual LAN ID (PVID) is used, the packets are not tagged, and a single internal network is connected to a switch. There are no virtual local area networks (VLAN) tagged ports set up on the Ethernet switch, and all virtual Ethernet adapters are defined using a single default PVID and no additional VLAN IDs (VIDs).

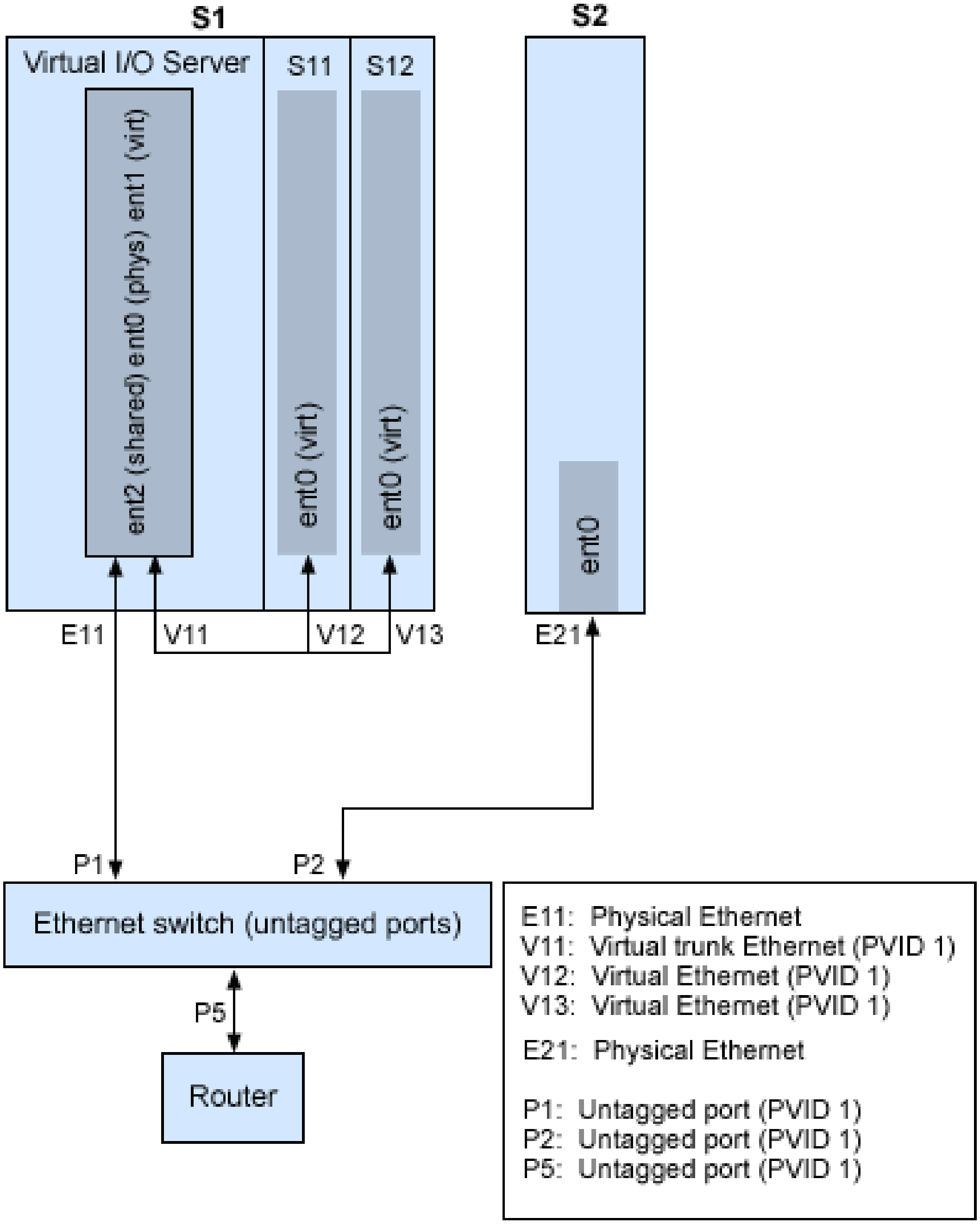
Prerequisites and assumptions

- The Hardware Management Console (HMC) was set up. To view the PDF file of Installing and configuring the Hardware Management Console, approximately 3 MB in size, see http://publib.boulder.ibm.com/infocenter/systems/scope/hw/topic/iphai/iphai.pdf .
- You understand the partitioning concepts as described in the Logical partitioning. To view the PDF file of Logical partitioning, approximately 1 MB in size, see http://publib.boulder.ibm.com/infocenter/systems/scope/hw/topic/iphat/iphat.pdf .
- The Virtual I/O Server logical partition has been created and the Virtual I/O Server has been installed. For instructions, see "Installing the Virtual I/O Server and client logical partitions" on page 78.
- You have created the remaining logical partitions that you want added to the network configuration.
- You have an Ethernet switch and a router ready to add to the configuration.
- You have IP addresses for all logical partitions and systems that will be added to the configuration.

While this procedure describes configuration in an HMC environment, this configuration is also possible in an Integrated Virtualization Manager environment.

Configuration steps

The following figure shows the configuration that will be completed during this scenario.


S1
Virtual I/O Server
S11
S12
S2
ent2 (shared) ent0 (phys) ent1 (virt)
ent0 (virt)
ent0 (virt)
ent0
E11
V11
V12
V13
E21
P1
P2
Ethernet switch (untagged ports)
P5
Router
E11: Physical Ethernet
V11: Virtual trunk Ethernet (PVID 1)
V12: Virtual Ethernet (PVID 1)
V13: Virtual Ethernet (PVID 1)
E21: Physical Ethernet
P1: Untagged port (PVID 1)
P2: Untagged port (PVID 1)
P5: Untagged port (PVID 1)

Using the preceding figure as a guide, follow these steps:

1. Set up an Ethernet switch with untagged ports. Alternatively, you can use an Ethernet switch that does not use VLAN.
2. For system S1, use the HMC to create a virtual Ethernet adapter (V11) for the Virtual I/O Server with the trunk setting, PVID set to 1, and no additional VIDs.
3. For system S1, use the HMC to create virtual Ethernet adapters V12 and V13 for logical partitions S11 and S12, respectively, with PVID set to 1 and no additional VIDs.
4. For system S1, use the HMC to assign physical Ethernet adapter E11 to the Virtual I/O Server and connect the adapter to the Ethernet switch port P1.

5. On the Virtual I/O Server, set up Shared Ethernet Adapter ent2 with the physical adapter ent0 and virtual adapter ent1.
6. Start the logical partitions. The process recognizes the virtual devices that were created in Step 1.
7. Configure IP addresses for S11 (en0), S12 (en0), and S2 (en0), so that they all belong to the same subnet with the router connected to Ethernet switch port P5.

The Shared Ethernet Adapter on the Virtual I/O Server logical partition can also be configured with IP addresses on the same subnet. This is required only for network connectivity to the Virtual I/O Server logical partition.

Scenario: Configuring a Virtual I/O Server using VLAN tagging

Use this scenario to help you become familiar with creating a network using VLAN tagging.

Situation

You are the system administrator responsible for planning and configuring the network in an environment with the Virtual I/O Server running. You would like to configure the network so that two logical subnets exist, with some logical partitions on each subnet.

Objective

The objective of this scenario is to configure multiple networks to share a single physical Ethernet adapter. Systems on the same subnet are required to be on the same VLAN and therefore have the same VLAN ID, which allows communication without having to go through the router. The separation in the subnets is achieved by ensuring that the systems on the two subnets have different VLAN IDs.

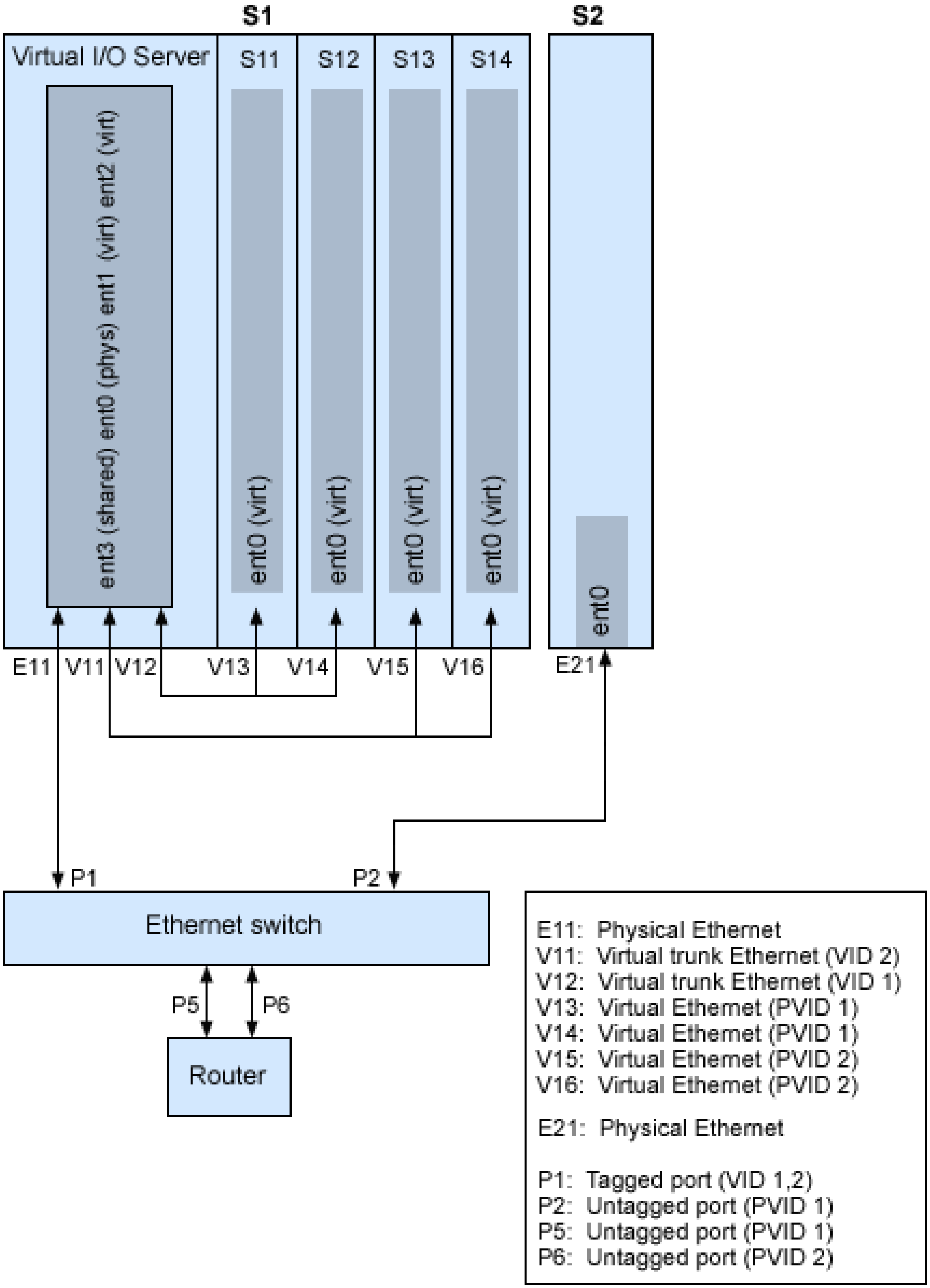
Prerequisites and assumptions

- The Hardware Management Console (HMC) was set up. To view the PDF file of Installing and configuring the Hardware Management Console, approximately 3 MB in size, see http://publib.boulder.ibm.com/infocenter/systems/scope/hw/topic/iphai/iphai.pdf .
- You understand the partitioning concepts as described in the Logical partitioning. To view the PDF file of Logical partitioning, approximately 1 MB in size, see http://publib.boulder.ibm.com/infocenter/systems/scope/hw/topic/iphat/iphat.pdf .
- The Virtual I/O Server logical partition has been created and the Virtual I/O Server has been installed. For instructions, see “Installing the Virtual I/O Server and client logical partitions” on page 78.
- You have created the remaining AIX or Linux logical partitions that you want added to the network configuration. (You cannot use VLAN tagging with IBM i logical partitions.)
- You have an Ethernet switch and a router ready to add to the configuration.
- You have IP addresses for all logical partitions and systems that will be added to the configuration.

You cannot use VLAN in an Integrated Virtualization Manager environment.

Configuration steps

The following figure shows the configuration that will be completed during this scenario.


S1
S2
Virtual I/O Server
S11
S12
S13
S14
ent3 (shared) ent0 (phys) ent1 (virt) ent2 (virt)
ent0 (virt)
ent0 (virt)
ent0 (virt)
ent0 (virt)
ent0
E11
V11
V12
V13
V14
V15
V16
E21
P1
P2
Ethernet switch
P5
P6
Router
E11: Physical Ethernet
V11: Virtual trunk Ethernet (VID 2)
V12: Virtual trunk Ethernet (VID 1)
V13: Virtual Ethernet (PVID 1)
V14: Virtual Ethernet (PVID 1)
V15: Virtual Ethernet (PVID 2)
V16: Virtual Ethernet (PVID 2)
E21: Physical Ethernet
P1: Tagged port (VID 1,2)
P2: Untagged port (PVID 1)
P5: Untagged port (PVID 1)
P6: Untagged port (PVID 2)

Using the preceding figure as a guide, follow these steps.

1. Set up the Ethernet switch ports as follows:
   - P1: Tagged port (VID 1, 2)

   - P2: Untagged port (PVID 1)
   - P5: Untagged port (PVID 1)
   - P6: Untagged port (PVID 2)

   For instructions on configuring the ports, see the documentation for your switch.
2. For system S1, use the HMC to create virtual Ethernet adapters for the Virtual I/O Server:
   - Create virtual Ethernet adapter V11 for the Virtual I/O Server with the trunk setting selected and VID set to 2. Specify an unused PVID value. This value is required, even though it will not be used.
   - Create virtual Ethernet adapter V12 for the Virtual I/O Server with the trunk setting selected and VID set to 1. Specify an unused PVID value. This value is required, even though it will not be used.
3. For system S1, use the HMC to create virtual Ethernet adapters for other logical partitions:
   - Create virtual adapters V13 and V14 for logical partitions S11 and S12, respectively, with PVID set to 2 and no additional VIDs.
   - Create virtual adapters V15 and V16 for logical partitions S13 and S14, respectively, with PVID set to 1 and no additional VIDs.
4. For system S1, use the HMC to assign the physical Ethernet adapter (E11) to the Virtual I/O Server and connect the adapter to the Ethernet switch port P1.
5. Using the Virtual I/O Server command-line interface, set up a Shared Ethernet Adapter ent3 with the physical adapter ent0 and virtual adapters ent1 and ent2.
6. Configure IP addresses for the following:
   - S13 (en0), S14 (en0), and S2 (en0) belong to VLAN 1 and are on the same subnet. The router is connected to Ethernet switch port P5.
   - S11 (en0) and S12 (en0) belong to VLAN 2 and are on the same subnet. The router is connected to Ethernet switch port P6.

You can configure the Shared Ethernet Adapter on the Virtual I/O Server logical partition with an IP address. This is required only for network connectivity to the Virtual I/O Server.

As the tagged VLAN network is being used, you must define additional VLAN devices over the Shared Ethernet Adapters before configuring IP addresses.

Scenario: Configuring Shared Ethernet Adapter failover

Use this article to help you become familiar with typical Shared Ethernet Adapter failover scenario.

Situation

You are the system administrator responsible for planning and configuring the network in an environment with the Virtual I/O Server running. You want to provide higher network availability to the client logical partition on the system. This can be accomplished by configuring a backup Shared Ethernet Adapter in a different Virtual I/O Server logical partition.

Objective

The objective of this scenario is to configure primary and backup Shared Ethernet Adapters in the Virtual I/O Server logical partitions so that network connectivity in the client logical partitions will not be lost in the case of adapter failure.

Prerequisites and assumptions

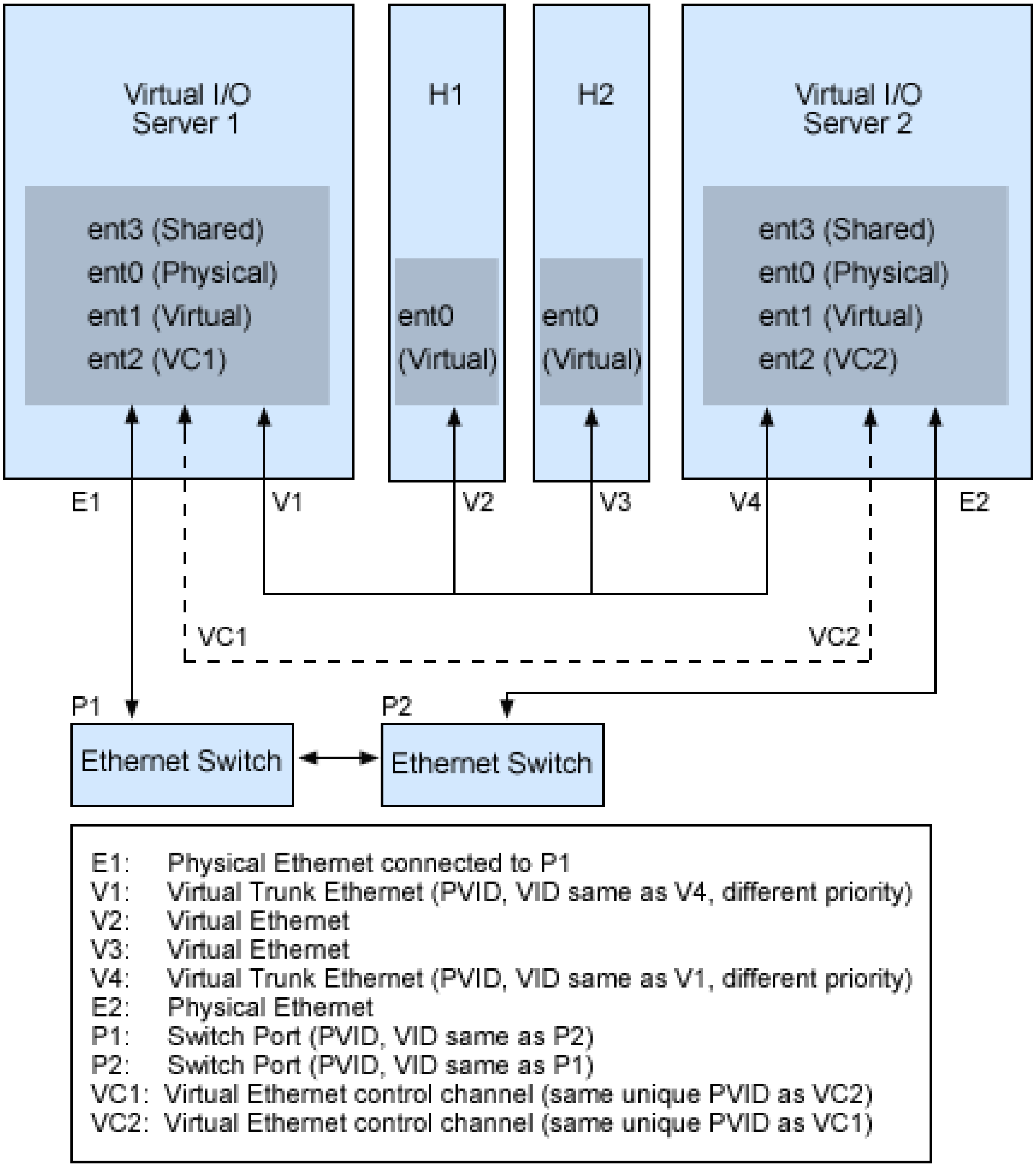
- The Hardware Management Console (HMC) was set up. To view the PDF file of Installing and configuring the Hardware Management Console, approximately 3 MB in size, see http://publib.boulder.ibm.com/infocenter/systems/scope/hw/topic/iphai/iphai.pdf .

- You understand the partitioning concepts as described in the Logical partitioning. To view the PDF file of Logical partitioning, approximately 1 MB in size, see

  http://publib.boulder.ibm.com/infocenter/systems/scope/hw/topic/iphat/iphat.pdf .
- Two separate Virtual I/O Server logical partitions have been created and the Virtual I/O Server has been installed in each logical partition. For instructions, see “Installing the Virtual I/O Server and client logical partitions” on page 78.
- You understand what Shared Ethernet Adapter failover is and how it works. See “Shared Ethernet Adapter failover” on page 72.
- You have created the remaining logical partitions that you want added to the network configuration.
- EachVirtual I/O Server logical partition has an available physical Ethernet adapter assigned to it.
- You have IP addresses for all logical partitions and systems that will be added to the configuration.

You cannot use the Integrated Virtualization Manager with multiple Virtual I/O Server logical partitions on the same server.

The following image depicts a configuration where the Shared Ethernet Adapter failover feature is set up. The client logical partitions H1 and H2 are accessing the physical network using the Shared Ethernet Adapters, which are the primary adapters. The virtual Ethernet adapters used in the shared Ethernet setup are configured with the same VLAN membership information (PVID, VID), but have different priorities. A dedicated virtual network forms the control channel and is required to facilitate communication between the primary and backup shared Ethernet device.


Virtual I/O Server 1
ent3 (Shared)
ent0 (Physical)
ent1 (Virtual)
ent2 (VC1)
H1
ent0 (Virtual)
H2
ent0 (Virtual)
Virtual I/O Server 2
ent3 (Shared)
ent0 (Physical)
ent1 (Virtual)
ent2 (VC2)
E1
V1
V2
V3
V4
E2
VC1
VC2
P1
P2
Ethernet Switch
Ethernet Switch
E1: Physical Ethernet connected to P1
V1: Virtual Trunk Ethernet (PVID, VID same as V4, different priority)
V2: Virtual Ethernet
V3: Virtual Ethernet
V4: Virtual Trunk Ethernet (PVID, VID same as V1, different priority)
E2: Physical Ethernet
P1: Switch Port (PVID, VID same as P2)
P2: Switch Port (PVID, VID same as P1)
VC1: Virtual Ethernet control channel (same unique PVID as VC2)
VC2: Virtual Ethernet control channel (same unique PVID as VC1)

Using the preceding figure as a guide, follow these steps:

1. On the HMC, create the virtual Ethernet adapters following these guidelines:
   - Configure the virtual adapters to be used for data as trunk adapters by selecting the trunk setting.
   - Assign different prioritization values (valid values are 1-15) to each virtual adapter.
   - Configure another virtual Ethernet to be used for the control channel by giving it a unique PVID value. Make sure you use the same PVID when creating this virtual Ethernet for both Virtual I/O Server logical partitions.
2. Using the Virtual I/O Server command line, run the following command to configure the Shared Ethernet Adapter. Run this command on both Virtual I/O Server logical partitions involved in the configuration:

```
mkvdev -sea physical_adapter -vadapter virtual_adapter -default
virtual_adapter\
-defaultid PVID_of_virtual_adapter -attr ha_mode=auto
ctl_chan=control_channel_adapter
```

For example, in this scenario, we ran the following command on both Virtual I/O Server logical partitions:

```
mkvdev -sea ent0 -vadapter ent1 -default ent1 -defaultid 60 -attr ha_mode=auto
ctl_chan=ent2
```

Scenario: Configuring Network Interface Backup in AIX client logical partitions without VLAN tagging

Use this scenario to become familiar with using a Network Interface Backup configuration in Virtual I/O clients that are running AIX logical partitions and are not configured for VLAN tagging.

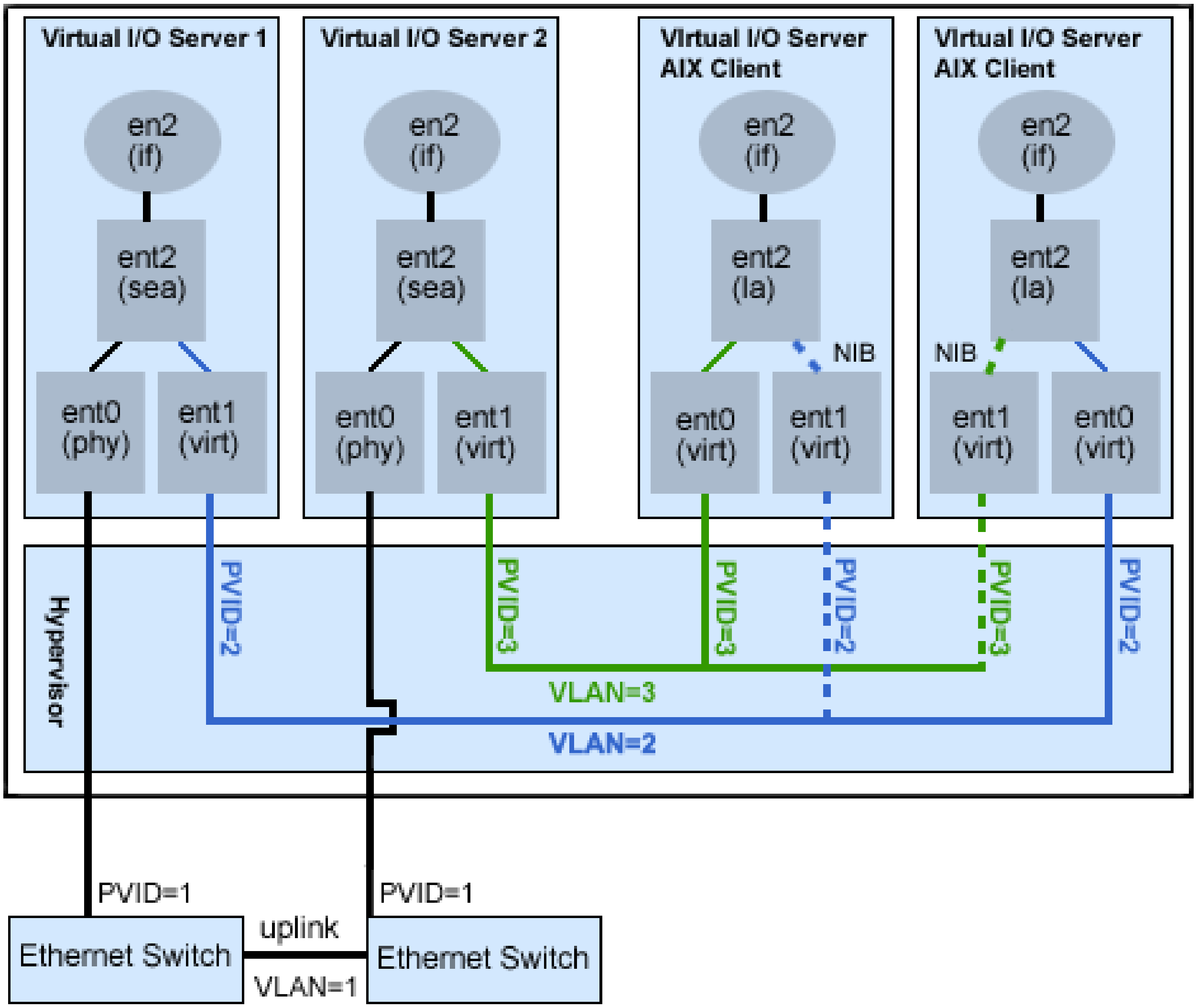
Situation

In this scenario, you want to configure a highly available virtual environment for your bridged network using the Network Interface Backup (NIB) approach to access external networks from your Virtual I/O clients. You do not plan to use VLAN tagging in your network setup. This approach requires you to configure a second Ethernet adapter on a different VLAN for each client and requires a Link Aggregation adapter with NIB features. This configuration is available for AIX logical partitions.

Typically, a Shared Ethernet Adapter failover configuration is the recommended configuration for most environments because it supports environments with or without VLAN tagging. Also, the NIB configuration is more complex than a Shared Ethernet Adapter failover configuration because it must be implemented on each of the clients. However, Shared Ethernet Adapter failover was not available prior to version 1.2 of Virtual I/O Server, and NIB was the only approach to a highly available virtual environment. Also, you might consider that in an NIB configuration you can distribute clients over both Shared Ethernet Adapters in such a way that half of them will use the first Shared Ethernet Adapter and the other half will use the second Shared Ethernet Adapter as primary adapter.

Objective

Create a virtual Ethernet environment using a Network Interface Backup configuration as depicted in the following figure.



Prerequisites and assumptions

Before completing the configuration tasks, review the following prerequisites and assumptions.

- The Hardware Management Console (HMC) is already set up. To view the PDF file of Installing and configuring the Hardware Management Console, approximately 3 MB in size, see http://publib.boulder.ibm.com/infocenter/systems/scope/hw/topic/iphai/iphai.pdf .
- Two separate Virtual I/O Server logical partitions have been created and the Virtual I/O Server has been installed in each logical partition. See the instructions in "Installing the Virtual I/O Server and client logical partitions" on page 78.
- You have created the remaining logical partitions that you want added to the network configuration.
- Each Virtual I/O Server logical partition has an available physical Ethernet adapter assigned to it.
- You have IP addresses for all logical partitions and systems that will be added to the configuration.

Configuration tasks

Using the figure as a guide, complete the following tasks to configure the NIB virtual environment.

1. Create a LAN connection between the Virtual I/O Servers and the external network:

   a. Configure a Shared Ethernet Adapter on the primary Virtual I/O Server that bridges traffic between the virtual Ethernet and the external network. See "Configuring a Shared Ethernet Adapter" on page 105.
   b. Configure a Shared Ethernet Adapter on the second Virtual I/O Server, as in step 1.
2. For each client logical partition, use the HMC to create a virtual Ethernet whose PVID matches the PVID of the primary Virtual I/O Server. This will be used as the primary adapter.
3. For each client logical partition, use the HMC to create a second virtual Ethernet whose PVID matches the PVID of the second (backup) Virtual I/O Server. This will be used as the backup adapter.
4. Create the Network Interface Backup setup using a Link Aggregation configuration. To create this configuration, follow the procedure Configuring an EtherChannel in the IBM System p and AIX Information Center. Make sure that you specify the following items:
   a. Select the primary Ethernet Adapter.
   b. Select the Backup Adapter.
   c. Specify the Internet Address to Ping. Select the IP address or hostname of a host outside of the Virtual I/O Server system that NIB will continuously ping to detect Virtual I/O Server failure.

Note: Keep in mind, when you configure NIB with two virtual Ethernet adapters, the internal networks used must stay separated in the hypervisor. You must use different PVIDs for the two adapters in the client and cannot use additional VIDs on them.

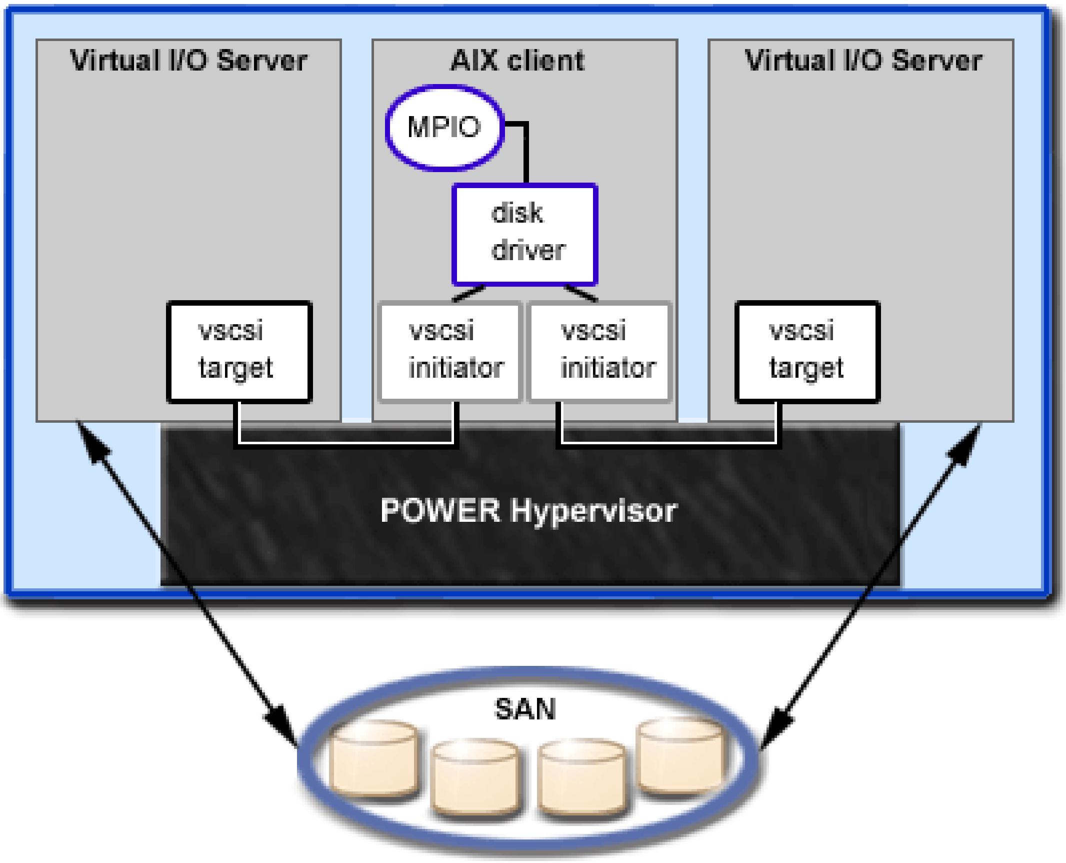
Scenario: Configuring Multi-Path I/O for AIX client logical partitions

Multi-Path I/O (MPIO) helps provide increased availability of virtual SCSI resources by providing redundant paths to the resource. This topic describes how to set up Multi-Path I/O for AIX client logical partitions.

In order to provide MPIO to AIX client logical partitions, you must have two Virtual I/O Server logical partitions configured on your system. This procedure assumes that the disks are already allocated to both the Virtual I/O Server logical partitions involved in this configuration.

To configure MPIO, follow these steps. In this scenario, hdisk5 in the first Virtual I/O Server logical partition, and hdisk7 in the second Virtual I/O Server logical partition, are used in the configuration.

The following figure shows the configuration that will be completed during this scenario.



Using the preceding figure as a guide, follow these steps:

1. Using the HMC, create SCSI server adapters on the two Virtual I/O Server logical partitions.
2. Using the HMC, create two virtual client SCSI adapters on the client logical partitions, each mapping to one of the Virtual I/O Server logical partitions.
3. On either of the Virtual I/O Server logical partitions, determine which disks are available by typing `lsdev -type disk`. Your results look similar to the following:

   ```
   name            status     description

   hdisk3          Available  MPIO Other FC SCSI Disk Drive
   hdisk4          Available  MPIO Other FC SCSI Disk Drive
   hdisk5          Available  MPIO Other FC SCSI Disk Drive
   ```

   Select which disk that you want to use in the MPIO configuration. In this scenario, we selected hdisk5.
4. Determine the ID of the disk that you have selected. For instructions, see "Identifying exportable disks" on page 102. In this scenario, the disk does not have an IEEE volume attribute identifier or a unique identifier (UDID), so we determine the physical identifier (PVID) by running the `lspv hdisk5` command. Your results look similar to the following:

   ```
   hdisk5          00c3e35ca560f919                    None
   ```

   The second value is the PVID. In this scenario, the PVID is 00c3e35ca560f919. Note this value.
5. List the attributes of the disk using the **lsdev** command. In this scenario, we typed `lsdev -dev hdisk5 -attr`. Your results look similar to the following

   ```
   ..
   lun_id          0x5463000000000000              Logical Unit Number ID          False
   ..
   ```

```
..
pvid            00c3e35ca560f9190000000000000000 Physical volume identifier       False
..
reserve_policy  single_path                      Reserve Policy                   True
```

Note the values for lun_id and reserve_policy. If the reserve_policy attribute is set to anything other than `no_reserve`, then you must change it. Set the reserve_policy to `no_reserve` by typing `chdev -dev hdiskx -attr reserve_policy=no_reserve`.

6. On the second Virtual I/O Server logical partition, list the physical volumes by typing `lspv`. In the output, locate the disk that has the same PVID as the disk identified previously. In this scenario, the PVID for hdisk7 matched:

   ```
   hdisk7          00c3e35ca560f919                    None
   ```

   Tip: Although the PVID values should be identical, the disk numbers on the two Virtual I/O Server logical partitions might vary.

7. Determine if the reserve_policy attribute is set to no_reserve using the **lsdev** command. In this scenario, we typed `lsdev -dev hdisk7 -attr`. You see results similar to the following:

   ```
   ..
   lun_id          0x5463000000000000               Logical Unit Number ID           False
   ..
   pvid            00c3e35ca560f9190000000000000000 Physical volume identifier       False
   ..
   reserve_policy  single_path                      Reserve Policy
   ```

   If the reserve_policy attribute is set to anything other than `no_reserve`, you must change it. Set the reserve_policy to `no_reserve` by typing `chdev -dev hdiskx -attr reserve_policy=no_reserve`.

8. On both Virtual I/O Server logical partitions, use the **mkvdev** to create the virtual devices. In each case, use the appropriate hdisk value. In this scenario, we type the following commands:
   - On the first Virtual I/O Server logical partition, we typed `mkvdev -vdev hdisk5 -vadapter vhost5 -dev vhdisk5`
   - On the second Virtual I/O Server logical partition, we typed `mkvdev -vdev hdisk7 -vadapter vhost7 -dev vhdisk7`

   The same LUN is now exported to the client logical partition from both Virtual I/O Server logical partitions.

9. AIX can now be installed on the client logical partition. For instructions on installing AIX, see Installing AIX in a Partitioned Environment in the IBM System p and AIX Information Center.

10. After you have installed AIX on the client logical partition, check for MPIO by running the following command:

    ```
    lspath
    ```

    You see results similar to the following:

    ```
    Enabled hdisk0 vscsi0
    Enabled hdisk0 vscsi1
    ```

    If one of the Virtual I/O Server logical partitions fails, the results of the lspath command look similar to the following:

    ```
    Failed  hdisk0 vscsi0
    Enabled hdisk0 vscsi1
    ```

    Unless a health check is enabled, the state continues to show `Failed` even after the disk has recovered. To have the state updated automatically, type `chdev -l hdiskx -a hcheck_interval=60 -P`. The client logical partition must be rebooted for this change to take effect.

Planning for the Virtual I/O Server

Use this topic to help gain an understanding of what to consider when planning for the Virtual I/O Server. In this section, you will find information about planning for the Virtual I/O Server.

Planning for Virtual I/O Server and client logical partitions using system plans

You can use the System Planning Tool (SPT) to create a system plan that includes configuration specifications for a Virtual I/O Server and client logical partitions. You can also use the Hardware Management Console (HMC) to create a system plan based on an existing system configuration.

SPT is a PC-based browser application that can assist you in planning and designing a new system. SPT validates your plan against system requirements and prevents you from creating a plan that exceeds those requirements. It also incorporates the IBM Systems Workload Estimator (WLE) to help you plan for workloads and performance. The output is a system-plan file that you can deploy to a managed system.

With SPT version 3.0 and later, you can include configuration specifications for the following components of a Virtual I/O Server logical partition in your system plan.

Table 15. Networking and storage components included in system plans

Networking components	Storage components
• Backup virtual Ethernet adapters • EtherChannel or Link Aggregation devices • Shared Ethernet Adapter failover • Shared Ethernet Adapters • Virtual Ethernet adapter mappings between the Virtual I/O Server and its client logical partitions • Virtual Ethernet adapters • Virtual LANs	• Mirroring • Multi-path I/O • SAN volumes • Storage pools (Volume groups) • Virtual SCSI adapter mappings between the Virtual I/O Server and its client logical partitions • Virtual SCSI adapters

With SPT version 3.0 and later, you can include AIX and Linux installation information for client logical partitions in the system plan. For more information, see “Installing operating environments from a system plan by using the HMC” on page 53.

SPT currently does not help you plan for high availability on client logical partitions or Redundant Array of Independent Disks (RAID) solutions for the Virtual I/O Server. For planning information about RAID and high availability, see “RAID” on page 74 and “High Availability Cluster Multi-Processing” on page 71.

To create a system plan, complete one of the following tasks:

- Create a system plan by using SPT. For instructions, see the System Planning Tool Web site. With SPT, you can create a system plan that includes the following information:
  – Configuration specifications for a Virtual I/O Server logical partition
  – Configuration specifications for AIX, IBM i, and Linux client logical partitions.
  – Installation information for the Virtual I/O Server operating environment.
  – Installation information for AIX and Linux operating environments.
- Create a system plan based on an existing system configuration by using the HMC. For instructions, see “Creating a system plan by using the HMC” on page 55.

  Alternatively, you can use the mksysplan command to create a system plan based on an existing system configuration. The mksysplan command is available from the HMC.

After you have created a system plan, you can deploy the system plan to the managed system. System plans can be deployed to a system managed by the HMC. For instructions, see "Deploying a system plan by using the HMC" on page 82.

The HMC must be at version 7, or later, to deploy the Virtual I/O Server logical partition and operating environment, and it must be at V7R3.3.0, or later, to deploy AIX and Linux operating environments on client logical partitions. When you deploy the system plan, the HMC automatically performs the following tasks based on the information provided in the system plan:

- Creates the Virtual I/O Server logical partition and logical partition profile.
- Installs the Virtual I/O Server operating environment and provisions virtual resources.
- Creates the client logical partitions and logical partition profiles.
- Installs the AIX and Linux operating environments on client logical partitions.

When you deploy the Virtual I/O Server logical partition to a new system, or to a system that does not already have a Virtual I/O Server logical partition configured, you must deploy the Virtual I/O Server logical partition in its entirety, including provisioning items, such as Shared Ethernet Adapters, EtherChannel adapters (or Link Aggregation devices), storage pools, and backing devices. If the HMC is at version V7R3.3.0, or later, you can deploy a system plan that includes additional provisioning items to an existing logical partition on the managed server, as long as the items and the system plan itself meet all the appropriate validation requirements. For more detailed information about the restrictions that apply, see "System plan validation for the HMC" on page 56.

Related information

Introduction to Virtualization
This publication provides system operators and administrators with overview information about virtualization technologies and management tools.

Installing operating environments from a system plan by using the HMC

You can use the Hardware Management Console (HMC) version 7 to install an operating environment on a logical partition when you deploy a system plan.

For HMCs prior to HMC V7R3.3.0, you can deploy a system plan to install only the Virtual I/O Server operating environment. Beginning with HMC V7R3.3.0, you also can deploy a system plan to install AIX or Linux operating environments on logical partitions in a system plan.

Note: You can create a system plan with installation information for an AIX or Linux operating environment only in the System Planning Tool (SPT). If a system plan has installation information for an AIX or Linux operating environment, you still can deploy the system plan to systems that are managed by an earlier version of the HMC. Earlier versions of the HMC Deploy System Plan Wizard can deploy the logical partitions in the system plan and ignore any operating environment installation information in the system plan. The earlier versions of the HMC can deploy all other aspects of the system plan successfully, as long as the other items in the system plan are validated successfully.

System plans that contain AIX or Linux installation information can be deployed only to new logical partitions or to logical partitions that do not already have an operating environment installed on them. If the logical partition already has an operating environment installed, the HMC does not deploy the operating environment that the system plan specifies for that logical partition.

If you plan to deploy a system plan that includes the installation of an operating environment for a logical partition, ensure that the **Power off the system after all the logical partitions are powered off** attribute for the managed system is not selected. If this attribute is selected, system plan deployment will fail because the deployment process starts partitions and then powers off partitions as part of installing operating environments. Consequently, the managed system will power off during deployment when the deployment process powers off the partitions. To verify the setting for this system attribute, complete these steps:

1. In the HMC navigation area, select **Systems Management** → **Servers**.
2. In the Tasks area, click **Properties**. The Properties window for the selected managed system opens.
3. On the **General** tab, verify that the **Power off the system after all the logical partitions are powered off** attribute is not selected, and click **OK**.

Supported operating environments

The wizard provides support for installing the following operating environments:

- AIX: Version 5.3 or 6.1
- Red Hat Enterprise Linux: Support is provided for any of the following versions:
  - Red Hat Enterprise Linux EL-AS: Version 4, 4 QUI, 4 QU2, 4 QU3, 4QU4, 4.5, or 4.6
  - Red Hat Enterprise Linux EL-Server: Version 5 or version 5.1
- SUSE Linux Enterprise Server: Version 10, 10 SP1, 9, 9 SP1, 9 SP2, 9 SP3, or 9 SP4
- Virtual I/O Server: Version 1.5 and 1.5.2

Using customized automatic installation files for operating environment installation

When you deploy a system plan that contains operating environment installation information, you can use the Deploy System Plan Wizard to specify the resource location that the wizard needs for installing the operating system environment. You also can specify or change operating environment installation settings. However, you cannot use the wizard to create or edit any automatic installation files that might be specified for an operating environment installation.

If you want to use a customized automatic installation file with an operating environment in the system plan, you must create or obtain the necessary file and import the file into the System Planning Tool (SPT). You can then use the SPT to edit the file, if necessary, and to associate the customized file with the system plan for installation of the appropriate operating environment on a logical partition. These automatic installation files, which allow you to provide specialized installation settings, include kickstart files for Red Hat Enterprise Linux, AutoYaST files for SUSE Linux Enterprise Server, and BOSinit.data files for AIX. For example, you might want to create a customized automatic installation file with the necessary setting so that the operating environment is installed specific virtualized resource that is provided to a client partition by the Virtual I/O Server partition over a virtual SCSI connection.

Required information for installing operating environments

During the Customize Operating Environment Install step of the Deploy System Plan Wizard, you provide required resource information for the operating environment installation and make any needed changes to installation settings. This step does not occur if the plan does not contain operating environment installation information. This step includes the following configuration options for installing an operating environment on the target logical partition:

- Operating Environment Install Image Resource. This configuration option allows you to specify an existing location for the operating environment installation files. You also can choose to create a new resource location for the installation files that you need to install an operating environment.
- Modify Install Settings. This configuration option allows you to provide or change late-binding installation settings for the target logical partition of the operating environment installation. In almost all cases, you need to update a number of late-binding installation settings before you can deploy an operating environment on a logical partition. Late-binding installation settings are those settings that are specific to an individual installation instance, for example, the IP address and subnet mask for the target logical partition on which the operating environment is to be deployed. You also can view settings for any custom automatic installation files that are included with the system plan. However, you cannot change these settings. You can use custom installation files to customize the installation of

an operating environment during the system plan deployment process only if the system plan already contains the necessary files. You can create these files and associate them with a system plan only in the System Planning Tool (SPT).

- Save Modified Operating Environment Install Information. This configuration option allows you to save any changes that you make to late-binding operating environment installation settings. You can save changes to the current system-plan file, or you can save them to a new system-plan file.

Creating a system plan by using the HMC

You can use the Hardware Management Console (HMC) to create a new system plan, based on an existing system configuration, and then deploy that system plan to other managed systems.

When you create a system plan on the HMC, you can deploy the resulting system plan to create identical logical partition configurations on managed systems with identical hardware. The system plan contains specifications for the logical partitions and partition profiles of the managed system that you used as the basis of creating the system plan.

If you use HMC V7R3.3.0 or later to create the system plan, the system plan can also include operating environment information for a logical partition. You still can deploy the system plan to systems that are managed by an earlier version of the HMC. Earlier versions of the HMC Deploy System Plan Wizard can deploy the logical partitions in the system plan and ignore any operating environment installation information in the system plan.

Note: Although the system plan that you create by using HMC V7R3.3.0 or later can contain some information about AIX or Linux operating environments on logical partitions in the system plan, it does not contain the information needed to install those operating environments as part of deploying the system plan. If you want a system plan to have the necessary information for installing an AIX or Linux operating environment, you need to use the System Planning Tool (SPT). You can use the SPT either to create a system plan or to convert a system plan that you create with the HMC to the format that the SPT uses and then change the system plan in the SPT. You also must use the SPT if you want to include specific automatic installation files with the system plan, or to customize any automatic installation files for the system plan.

The new system plan also can contain hardware information that the HMC is able to obtain from the selected managed system. However, the amount of hardware information that the HMC can capture for the new system plan varies based on the method that the HMC uses to gather the hardware information.

There are two methods that the HMC potentially can use: inventory gathering and hardware discovery. For example, when using inventory gathering, the HMC can detect virtual device configuration information for the Virtual I/O Server. Additionally, the HMC can use one or both of these methods to detect disk and tape information for IBM i logical partitions.

Ensure that you meet the requirements for using either or both of the inventory gathering and hardware discovery methods before you create your system plan. See System plan creation requirements for more information.

Creating a system plan

To create a system plan by using the Hardware Management Console, complete the following steps:

1. In the navigation area, select **System Plans**. The System Plans page opens.
2. In the Tasks area, select **Create System Plan**. The Create System Plan window opens.
3. Select the managed system that you want to use as the basis for the new system plan.
4. Enter a name and description for the new system plan.
5. Optional: For Hardware Management Console V7R3.3.2, or later, select whether you want to retrieve inactive and unallocated hardware resources. This option appears only if the managed system is capable of hardware discovery, and the option is selected by default.

Note: If you do not select the **Retrieve inactive and unallocated hardware resources** option, the HMC does not perform a new hardware discovery, but instead uses the data in the inventory cache on the system. The HMC still performs inventory gathering and retrieves hardware information for any active logical partitions on the managed server. The resulting new system plan contains hardware information from the inventory-gathering process, as well as hardware information from the hardware inventory cache on the system.

6. Optional: Select whether you want to view the system plan immediately after the HMC creates it.
7. Click **Create**.

Now that you have a new system plan, you can export the system plan, import it onto another managed system, and deploy the system plan to that managed system.

Note: As an alternative to the HMC Web user interface, you can use the mksysplan command on the HMC to create a system plan based on the configuration of an existing managed system.

System plan validation for the HMC

You deploy a system plan to a system that is managed by a Hardware Management Console (HMC) managed system by using the System Plan Deployment Wizard. The wizard validates the information in the system plan against the configuration of the managed system before beginning the deployment process.

The Deploy System Plan wizard validates the system plan prior to deployment to ensure that it can be deployed successfully. The wizard validates the system plan in two phases. The first phase of the validation process is hardware validation. During this phase, the wizard is validating that the processors, memory, and I/O adapters that are available on the managed system match or exceed those that the system plan specifies. The wizard also validates that the hardware placement on the managed system matches the hardware placement that the system plan specifies.

The second phase of the validation process is partition validation. During this phase, the wizard validates that the logical partitions on the managed system match those in the system plan. If the system plan contains provisioning information, the wizard also validates the provisioning items in the system plan to determine which items are deployable.

If any step in the partition validation process fails for the system plan, validation of the entire system plan fails.

Specifications

This topic defines the range of configuration possibilities, including the minimum number of resources needed and the maximum number of resources allowed.

To activate the Virtual I/O Server, the PowerVM Editions (or Advanced POWER® Virtualization) hardware feature is required. A logical partition with enough resources to share with other logical partitions is required. The following is a list of minimum hardware requirements that must be available to create the Virtual I/O Server.

Table 16. Resources that are required

Resource	Requirement
Hardware Management Console or Integrated Virtualization Manager	The HMC or Integrated Virtualization Manager is required to create the logical partition and assign resources.
Storage adapter	The server logical partition needs at least one storage adapter.
Physical disk	The disk must be at least 30 GB. This disk can be shared.
Ethernet adapter	If you want to route network traffic from virtual Ethernet adapters to a Shared Ethernet Adapter, you need an Ethernet adapter.

Table 16. Resources that are required (continued)

Resource	Requirement
Memory	For POWER6 processor-based systems, at least 768 MB of memory is required. For POWER5™ processor-based systems, at least 512 MB of memory is required.
Processor	At least 0.1 processor is required.

The following table defines the limitations for storage management.

Table 17. Limitations for storage management

Category	Limit
Volume groups	4096 per system
Physical volumes	1024 per volume group
Physical partitions	1024 per volume group
Logical volumes	1024 per volume group
Logical partitions	No limit

Limitations and restrictions

Learn about Virtual I/O Server configuration limitations.

Consider the following when implementing virtual SCSI:

- Virtual SCSI supports the following connection standards for backing devices: fibre channel, SCSI, SCSI RAID, iSCSI, SAS, SATA, USB, and IDE.
- The SCSI protocol defines mandatory and optional commands. While virtual SCSI supports all of the mandatory commands, not all of the optional commands are supported.
- There are performance implications when you use virtual SCSI devices. Because the client/server model is made up of layers of function, using virtual SCSI can consume additional processor cycles when processing I/O requests.
- The Virtual I/O Server is a dedicated logical partition, to be used only for Virtual I/O Server operations. Other applications cannot run in the Virtual I/O Server logical partition.
- If there is a resource shortage, performance degradation might occur. If a Virtual I/O Server is serving many resources to other logical partitions, ensure that enough processor power is available. In case of high workload across virtual Ethernet adapters and virtual disks, logical partitions might experience delays in accessing resources.
- Logical volumes and files exported as virtual SCSI disks are always configured as single path devices on the client logical partition.
- Logical volumes or files exported as virtual SCSI disks that are part of the root volume group (rootvg) are not persistent if you reinstall the Virtual I/O Server. However, they are persistent if you update the Virtual I/O Server to a new service pack. Therefore, before reinstalling the Virtual I/O Server, ensure that you back up the corresponding clients' virtual disks. When exporting logical volumes, it is best to export logical volumes from a volume group other than the root volume group. When exporting files, it is best to create file storage pools and the virtual media repository in a parent storage pool other than the root volume group.

Consider the following when implementing virtual adapters:

- Only Ethernet adapters can be shared. Other types of network adapters cannot be shared.
- IP forwarding is not supported on the Virtual I/O Server.
- The maximum number of virtual adapters can be any value from 2 to 65,536. However, if you set the maximum number of virtual adapters to a value higher than 1024, the logical partition might fail to activate or the server firmware might require more system memory to manage the virtual adapters.

The Virtual I/O Server supports client logical partitions running the following operating systems on the following POWER6 processor-based servers.

Table 18. Operating system support for Virtual I/O Server client logical partitions

Operating system	**POWER6 processor-based servers**
AIX 5.3 or later	All POWER6 processor-based servers
IBM i 6.1 or later	All POWER6 processor-based servers
SUSE Linux Enterprise Server 10 Service Pack 2 or later	• 9119-FHA • 9125-F2A
SUSE Linux Enterprise Server 10 Service Pack 1	• 8203-E4A • 8204-E8A • 9117-MMA • 9406-MMA • 9407-M15 • 9408-M25 • 9409-M50
Red Hat Enterprise Linux version 5.2	• 9119-FHA • 9125-F2A
Red Hat Enterprise Linux version 5.1	• 8203-E4A • 8204-E8A • 9117-MMA • 9406-MMA • 9407-M15 • 9408-M25 • 9409-M50
Red Hat Enterprise Linux version 4.7	9119-FHA
Red Hat Enterprise Linux version 4.6	9125-F2A
Red Hat Enterprise Linux version 4.5	• 8203-E4A • 8204-E8A • 9117-MMA • 9406-MMA • 9407-M15 • 9408-M25 • 9409-M50

The Virtual I/O Server supports client logical partitions running the following operating systems on POWER5 processor-based servers:

- AIX 5.3 (or later)
- SUSE Linux Enterprise Server 9 (or later)
- SUSE Linux Enterprise Server 10 (or later)
- Red Hat Enterprise Linux version 4 (or later)
- Red Hat Enterprise Linux version 5 (or later)

Capacity planning

This topic includes capacity-planning considerations for the Virtual I/O Server, including information about hardware resources and limitations.

Client logical partitions might use virtual devices, dedicated devices, or a combination of both. Before you begin to configure and install the Virtual I/O Server and client logical partitions, plan what resources each logical partition will use. Throughput requirements and overall workload must be considered when deciding whether to use virtual or dedicated devices and when allocating resources to the Virtual I/O Server. Compared to dedicated SCSI disks, virtual SCSI disks might achieve similar throughput numbers depending on several factors, including workload and virtual SCSI resources. However, virtual SCSI devices generally have higher processor utilization when compared with directly attached storage.

Planning for virtual SCSI

Find capacity-planning and performance information for virtual SCSI.

Different I/O subsystems have different performance qualities, as does virtual SCSI. This section discusses the performance differences between physical and virtual I/O. The following topics are described in this section:

Virtual SCSI latency:

Find information about virtual SCSI latency.

I/O latency is the amount of time that passes between the initiation and completion of a disk I/O operation. For example, consider a program that performs 1000 random disk I/O operations, one at a time. If the time to complete an average operation is 6 milliseconds, the program runs in no fewer than 6 seconds. However, if the average response time is reduced to 3 milliseconds, the run time might be reduced by 3 seconds. Applications that are multithreaded or use asynchronous I/O might be less sensitive to latency, but in most circumstances, lower latency can help improve performance.

Because virtual SCSI is implemented as a client and server model, there is some latency that does not exist with directly attached storage. The latency might range from 0.03 to 0.06 milliseconds per I/O operation depending primarily on the block size of the request. The average latency is comparable for both physical disk and logical volume-backed virtual drives. The latency experienced when using a Virtual I/O Server in a shared-processor logical partition can be higher and more variable than using a Virtual I/O Server in a dedicated logical partition. For additional information about the performance differences between dedicated logical partitions and shared-processor logical partitions, see “Virtual SCSI sizing considerations” on page 60.

The following table identifies latency (in milliseconds) for different block-size transmissions on both physical disk and logical-volume-backed virtual SCSI disks.

Table 19. Increase in disk I/O response time based on block size (in milliseconds)

Backing type	**4 K**	**8 K**	**32 K**	**64 K**	**128 K**
Physical disk	0.032	0.033	0.033	0.040	0.061
Logical volume	0.035	0.036	0.034	0.040	0.063

The average disk-response time increases as the block size increases. The latency increases for a virtual SCSI operation are relatively greater on smaller block sizes because of their shorter response time.

Virtual SCSI bandwidth:

View information about virtual SCSI bandwidth.

I/O bandwidth is the maximum amount of data that can be read or written to a storage device in a unit of time. Bandwidth can be measured from a single thread or from a set of threads running concurrently. Although many customer applications are more sensitive to latency than bandwidth, bandwidth is crucial for many typical operations, such as backing up and restoring persistent data.

The following table compares the results of bandwidth tests for virtual SCSI and physical I/O performance. In the tests, a single thread operates sequentially on a constant file that is 256 MB in size with a Virtual I/O Server running in a dedicated partition. More I/O operations are issued when reading or writing to the file using a small block size as compared to a larger block size. The test was conducted using a storage server with feature code 6239 (type 5704/0625) and a 2-gigabit Fibre Channel adapter attached to one RAID0 LUN that is composed of 5 physical disks from a DS4400 disk system (formerly a FAStT700). The table shows the comparison of measured bandwidth in megabytes per second (MB/s) using virtual SCSI and local attachment for reads with varying block sizes of operations. The difference between virtual I/O and physical I/O in these tests is attributable to the increased latency when using virtual I/O. Because of the larger number of operations, the bandwidth measured with small block sizes is lower than with large block sizes.

Table 20. Physical and virtual SCSI bandwidth comparison (in MB/s)

I/O type	4 K	8 K	32 K	64 K	128 K
Virtual	20.3	35.4	82.6	106.8	124.5
Physical	24.3	41.7	90.6	114.6	132.6

Virtual SCSI sizing considerations:

Understand the processor and memory-sizing considerations when implementing virtual SCSI .

When you are designing and implementing a virtual SCSI application environment, consider the following sizing issues:

- The amount of memory allocated to the Virtual I/O Server
- The processor entitlement of the Virtual I/O Server
- Whether the Virtual I/O Server is run as a shared-processor logical partition or as a dedicated processor logical partition
- The maximum transfer size limitation for physical devices and AIX clients

The processor impacts of using virtual I/O on the client are insignificant. The processor cycles run on the client to perform a virtual SCSI I/O operation are comparable to that of a locally attached I/O device. Thus, there is no increase or decrease in sizing on the client logical partition for a known task. These sizing techniques do not anticipate combining the function of shared Ethernet with the virtual SCSI server. If the two are combined, consider adding resources to account for the shared Ethernet activity with virtual SCSI.

Virtual SCSI sizing using dedicated processor logical partitions

The amount of processor entitlement required for a virtual SCSI server is based on the maximum I/O rates required of it. Because virtual SCSI servers do not normally run at maximum I/O rates all of the time, the use of surplus processor time is potentially wasted when using dedicated processor logical partitions. In the first of the following sizing methodologies, you need a good understanding of the I/O rates and I/O sizes required of the virtual SCSI server. In the second, we will size the virtual SCSI server based on the I/O configuration.

The sizing methodology used is based on the observation that the processor time required to perform an I/O operating on the virtual SCSI server is fairly constant for a given I/O size. It is a simplification to make this statement, because different device drivers have subtly varying efficiencies. However, under most circumstances, the I/O devices supported by the virtual SCSI server are sufficiently similar. The following table shows approximate cycles per second for both physical disk and logical volume operations on a 1.65 Ghz processor. These numbers are measured at the physical processor; simultaneous multithreading (SMT) operation is assumed. For other frequencies, scaling by the ratio of the frequencies (for example, 1.5 Ghz = 1.65 Ghz / 1.5 Ghz × cycles per operation) is sufficiently accurate to produce a reasonable sizing.

Table 21. Approximate cycles per second on a 1.65 Ghz logical partition

Disk type	**4 KB**	**8 KB**	**32 KB**	**64 KB**	**128 KB**
Physical disk	45,000	47,000	58,000	81,000	120,000
Logical volume	49,000	51,000	59,000	74,000	105,000

Consider a Virtual I/O Server that uses three client logical partitions on physical disk-backed storage. The first client logical partition requires a maximum of 7,000 8-KB operations per second. The second client logical partition requires a maximum of 10,000 8-KB operations per second. The third client logical partition requires a maximum of 5,000 128-KB operations per second. The number of 1.65 Ghz processors for this requirement is approximately ((7,000 × 47,000 + 10,000 × 47,000 + 5,000 × 120,000) / 1,650,000,000) = 0.85 processors, which rounds up to a single processor when using a dedicated processor logical partition.

If the I/O rates of the client logical partitions are not known, you can size the Virtual I/O Server to the maximum I/O rate of the storage subsystem attached. The sizing could be biased toward small I/O operations or large I/O operations. Sizing to maximum capacity for large I/O operations will balance the processor capacity of the Virtual I/O Server to the potential I/O bandwidth of the attached I/O. The negative aspect of this sizing methodology is that, in nearly every case, more processor entitlement will be assigned to the Virtual I/O Server than it will typically consume.

Consider a case in which a Virtual I/O Server manages 32 physical SCSI disks. An upper limit of processors required can be established based on assumptions about the I/O rates that the disks can achieve. If it is known that the workload is dominated by 8096-byte operations that are random, then assume that each disk is capable of approximately 200 disk I/O operations per second (15k rpm drives). At peak, the Virtual I/O Server would need to serve approximately 32 disks × 200 I/O operations per second × 47,000 cycles per operation, resulting in a requirement for approximately 0.19 processor performance. Viewed another way, a Virtual I/O Server running on a single processor should be capable of supporting more than 150 disks doing 8096-byte random I/O operations.

Alternatively, if the Virtual I/O Server is sized for maximum bandwidth, the calculation results in a higher processor requirement. The difference is that maximum bandwidth assumes sequential I/O. Because disks are more efficient when they are performing large, sequential I/O operations than they are when performing small, random I/O operations, a higher number of I/O operations per second can be performed. Assume that the disks are capable of 50 MB per second when doing 128 KB I/O operations. That situation implies each disk could average 390 disk I/O operations per second. Thus, the amount of processing power necessary to support 32 disks, each doing 390 I/O operations per second with an operation cost of 120,000 cycles (32 × 390 × 120,000 / 1,650,000,000) results in approximately 0.91 processors. Consequently, a Virtual I/O Server running on a single processor should be capable of driving approximately 32 fast disks to maximum throughput.

Virtual SCSI server sizing using shared processor logical partitions

Defining virtual SCSI servers in shared processor logical partitions allows more specific processor resource sizing and potential recovery of unused processor time by uncapped logical partitions. However, using shared-processor logical partitions for virtual SCSI servers can frequently increase I/O response time and make for somewhat more complex processor entitlement sizings.

The sizing methodology should be based on the same operation costs for dedicated logical partition I/O servers, with added entitlement for running in shared-processor logical partitions. Configure the Virtual I/O Server as uncapped, so that, if the Virtual I/O Server is undersized, there is opportunity to get more processor time to serve I/O operations.

Because I/O latency with virtual SCSI can vary due to a number of conditions, consider the following if a logical partition has high I/O requirements:

- Configure the logical partition with physical I/O if the configuration allows.
- In most cases, the Virtual I/O Server logical partition can use a shared, uncapped processor.

Virtual SCSI server memory sizing

Memory sizing in virtual SCSI is simplified because there is no caching of file data in the memory of the virtual SCSI server. Because there is no data caching, the memory requirements for the virtual SCSI server are fairly modest. With large I/O configurations and very high data rates, a 1 GB memory allocation for the virtual SCSI server is likely to be sufficient. For low I/O rate situations with a small number of attached disks, 512 MB will most likely suffice.

Virtual SCSI maximum transfer size limitation

If you add another virtual target device to the virtual SCSI server adapter and the new virtual target device has a smaller maximum transfer size than the other configured devices on that adapter, the Virtual I/O Server does not show a new virtual device to the client. At the time the virtual target device is created, the Virtual I/O Server displays a message stating that the new target device will not be visible to the client until you reboot the client.

To display the maximum transfer size of a physical device, use the following command: `lsdev -attr max_transfer -dev hdiskN`

Planning for Shared Ethernet Adapters

Use this section to find capacity-planning and performance information for Shared Ethernet Adapter. This section contains planning information and performance considerations for using Shared Ethernet Adapters on the Virtual I/O Server.

Network requirements:

This topic includes information you need in order to accurately size your Shared Ethernet Adapter environment.

To plan for using Shared Ethernet Adapters, you must determine your network needs. This section gives overview information of what should be considered when sizing the Shared Ethernet Adapter environment. Sizing the Virtual I/O Server for the Shared Ethernet Adapter involves the following factors:

- Defining the target bandwidth (MB per second), or transaction rate requirements (operations per second). The target performance of the configuration must be determined from your workload requirements.
- Defining the type of workload (streaming or transaction oriented).
- Identifying the maximum transmission unit (MTU) size that will be used (1500 or jumbo frames).
- Determining if the Shared Ethernet Adapter will run in a threaded or nonthreaded environment.
- Knowing the throughput rates that various Ethernet adapters can provide (see Adapter selection).
- Knowing the processor cycles required per byte of throughput or per transaction (see Processor allocation).

Bandwidth requirement

The primary consideration is determining the target bandwidth on the physical Ethernet adapter of the Virtual I/O Server. This will determine the rate that data can be transferred between the Virtual I/O Server and the client logical partitions. After the target rate is known, the correct type and number of network adapters can be selected. For example, Ethernet adapters of various speeds could be used. One or more adapters could be used on individual networks, or they could be combined using Link Aggregation (or EtherChannel).

Workload type

The type of workload to be performed must be considered, whether it is streaming of data for workloads such as file transfer, data backup, or small transaction workloads, such as remote procedure calls. The streaming workload consists of large, full-sized network packets and associated small, TCP acknowledgment packets. Transaction workloads typically involve smaller packets or might involve small requests, such as a URL, and a larger response, such as a Web page. A Virtual I/O Server will need to frequently support streaming and small packet I/O during various periods of time. In that case, approach the sizing from both models.

MTU size

The MTU size of the network adapters must also be considered. The standard Ethernet MTU is 1500 bytes. Gigabit Ethernet and 10 gigabit Ethernet can support 9000-byte MTU jumbo frames. Jumbo frames might reduce the processor cycles for the streaming types of workloads. However, for small workloads, the larger MTU size might not help reduce processor cycles.

Threaded or nonthreaded environment

Use threaded mode when virtual SCSI will be run on the same Virtual I/O Server logical partition as Shared Ethernet Adapter. Threaded mode helps ensure that virtual SCSI and the Shared Ethernet Adapter can share the processor resource appropriately. However, threading increases instruction-path length, which uses additional processor cycles. If the Virtual I/O Server logical partition will be dedicated to running shared Ethernet devices (and associated virtual Ethernet devices) only, the adapters should be configured with threading disabled. For more information, see "Processor allocation" on page 65.

Adapter throughput

Knowing the throughput capability of different Ethernet adapters can help you determine which adapters to use as Shared Ethernet Adapters and how many adapters to use. For more information, see "Adapter selection."

Processor entitlement

You must determine how much processor power is required to move data through the adapters at the desired rate. Networking device drivers are typically processor-intensive. Small packets can come in at a faster rate and use more processor cycles than larger packet workloads. Larger packet workloads are typically limited by network wire bandwidth and come in at a slower rate, thus requiring less processor power than small packet workloads for the amount of data transferred.

Adapter selection:

Use this section to find the attributes and performance characteristics of various types of Ethernet adapters to help you select which adapters to use in your environment.

This section provides approximate throughput rates for various Ethernet adapters set at various MTU sizes. Use this information to determine which adapters will be needed to configure a Virtual I/O Server. To make this determination, you must know the desired throughput rate of the client logical partitions.

Following are general guidelines for network throughput. These numbers are not specific, but they can serve as a general guideline for sizing. In the following tables, the 100 MB, 1 GB, and 10 GB speeds are rounded down for estimating.

Table 22. Simplex (one direction) streaming rates

Adapter speed	Approximate throughput rate
10 Mb Ethernet	1 MB/second
100 Mb Ethernet	10 MB/second
1000 Mb Ethernet (GB Ethernet)	100 MB/second
10000 Mb Ethernet (10 GB Ethernet, Host Ethernet Adapter or Integrated Virtual Ethernet)	1000 MB/second

Table 23. Full duplex (two direction) streaming rates on full duplex network

Adapter speed	Approximate throughput rate
10 Mb Ethernet	2 MB/second
100 Mb Ethernet	20 MB/second
1000 Mb Ethernet (Gb Ethernet)	150 MB/second
10000 Mb Ethernet (10 Gb Ethernet, Host Ethernet Adapter or Integrated Virtual Ethernet)	1500 MB/second

The following tables list maximum network payload speeds, which are user payload data rates that can be obtained by sockets-based programs for applications that are streaming data. The rates are a result of the network bit rate, MTU size, physical level overhead (such as interframe gaps and preamble bits), data link headers, and TCP/IP headers. A gigahertz-speed processor is assumed. These numbers are optimal for a single LAN. If your network traffic is going through additional network devices, your results might vary.

In the following tables, raw bit rate is the physical media bit rate and does not reflect interframe gaps, preamble bits, data link headers, and trailers. Interframe gaps, preamble bits, data link headers, and trailers can all reduce the effective usable bit rate of the wire.

Single direction (simplex) TCP streaming rates are rates that can be achieved by sending data from one machine to another in a memory-to-memory test. Full-duplex media can usually perform slightly better than half-duplex media because the TCP acknowledgment packets can flow without contending for the same wire that the data packets are flowing on.

Table 24. Single direction (simplex) TCP streaming rates

Network type	Raw bit rate (Mb)	Payload rate (Mb)	Payload rate (MB)
10 Mb Ethernet, Half Duplex	10	6	0.7
10 Mb Ethernet, Full Duplex	10 (20 Mb full duplex)	9.48	1.13
100 Mb Ethernet, Half Duplex	100	62	7.3
100 Mb Ethernet, Full Duplex	100 (200 Mb full duplex)	94.8	11.3
1000 Mb Ethernet, Full Duplex, MTU 1500	1000 (2000 Mb full duplex)	948	113
1000 Mb Ethernet, Full Duplex, MTU 9000	1000 (2000 Mb full duplex)	989	117.9

Table 24. Single direction (simplex) TCP streaming rates (continued)

Network type	Raw bit rate (Mb)	Payload rate (Mb)	Payload rate (MB)
1000 Mb Ethernet, Full Duplex, Host Ethernet Adapter (or Integrated Virtual Ethernet) MTU 1500	10000	9479	1130
1000 Mb Ethernet, Full Duplex, Host Ethernet Adapter (or Integrated Virtual Ethernet) MTU 9000	10000	9899	1180

Full-duplex TCP streaming workloads have data streaming in both directions. Workloads that can send and receive packets concurrently can take advantage of full duplex media. Some media, for example Ethernet in half-duplex mode, cannot send and receive concurrently, thus they will not perform any better, and can usually degrade performance, when running duplex workloads. Duplex workloads will not increase at a full doubling of the rate of a simplex workload because the TCP acknowledgment packets returning from the receiver must now compete with data packets flowing in the same direction.

Table 25. Two direction (duplex) TCP streaming rates

Network type	Raw bit rate (Mb)	Payload rate (Mb)	Payload rate (MB)
10 Mb Ethernet, Half Duplex	10	5.8	0.7
10 Mb Ethernet, Full Duplex	10 (20 Mb full duplex)	18	2.2
100 Mb Ethernet, Half Duplex	100	58	7
100 Mb Ethernet, Full Duplex	100 (200 Mb full duplex)	177	21.1
1000 Mb Ethernet, Full Duplex, MTU 1500	1000 (2000 Mb full duplex)	1470 (1660 peak)	175 (198 peak)
1000 Mb Ethernet, Full Duplex, MTU 9000	1000 (2000 Mb full duplex)	1680 (1938 peak)	200 (231 peak)
10000 Mb Ethernet, Host Ethernet Adapter (or Integrated Virtual Ethernet) Full Duplex, MTU 1500	10000	14680 (15099 peak)	1750 (1800 peak)
10000 Mb Ethernet, Host Ethernet Adapter (or Integrated Virtual Ethernet) Full Duplex, MTU 9000	10000	16777 (19293 pack)	2000 (2300 peak)

Note:

1. Peak numbers represent optimal throughput with multiple TCP sessions running in each direction. Other rates are for a single TCP session.
2. 1000 MB Ethernet (gigabit Ethernet) duplex rates are for the PCI-X adapter in PCI-X slots.
3. Data rates are for TCP/IP using the IPv4 protocol. Adapters with MTU set to 9000 have RFC 1323 enabled.

Processor allocation:

This section contains processor-allocation guidelines for both dedicated processor logical partitions and shared processor logical partitions.

Because Ethernet running MTU size of 1500 bytes consumes more processor cycles than Ethernet running Jumbo frames (MTU 9000), the guidelines are different for each situation. In general, the processor utilization for large packet workloads on jumbo frames is approximately half that required for MTU 1500.

If MTU is set to 1500, provide one processor (1.65 Ghz) per Gigabit Ethernet adapter to help reach maximum bandwidth. This equals ten 100-Mb Ethernet adapters if you are using smaller networks. For smaller transaction workloads, plan to use one full processor to drive the Gigabit Ethernet workload to maximum throughput. For example, if two Gigabit Ethernet adapters will be used, allocate up to two processors to the logical partition.

If MTU is set to 9000 (jumbo frames), provide 50% of one processor (1.65 Ghz) per Gigabit Ethernet adapter to reach maximum bandwidth. Small packet workloads should plan to use one full processor to drive the Gigabit Ethernet workload. Jumbo frames have no effect on the small packet workload case.

Shared Ethernet Adapter using a dedicated processor logical partition

The sizing provided is divided into two workload types: TCP streaming and TCP request and response. Both MTU 1500 and MTU 9000 networks were used in the sizing, which is provided in terms of machine cycles per byte of throughput for streaming or per transaction for request/response workloads.

The data in the following tables was derived using the following formula:

(number of processors × processor_utilization × processor clock frequency) / Throughput rate in bytes per second or transaction per second = cycles per Byte or transaction.

For the purposes of this test, the numbers were measured on a logical partition with one 1.65 Ghz processor with simultaneous multi-threading (SMT) enabled.

For other processor frequencies, the numbers in these tables can be scaled by the ratio of the processor frequencies for approximate values to be used for sizing. For example, for a 1.5 Ghz processor speed, use 1.65/1.5 × cycles per byte value from the table. This example would result in a value of 1.1 times the value in the table, thus requiring 10% more cycles to adjust for the 10% slower clock rate of the 1.5 Ghz processor.

To use these values, multiply your required throughput rate (in bytes or transactions) by the cycles per byte value in the following tables. This result will give you the required machine cycles for the workload for a 1.65 Ghz speed. Then adjust this value by the ratio of the actual machine speed to this 1.65 Ghz speed. To find the number of processors, divide the result by 1,650,000,000 cycles (or the cycles rate if you adjusted to a different speed machine). You would need the resulting number of processors to drive the workload.

For example, if the Virtual I/O Server must deliver 200 MB of streaming throughput, the following formula would be used:

200 × 1024 × 1024 × 11.2 = 2,348,810,240 cycles / 1,650,000,000 cycles per processor = 1.42 processors.

In round numbers, it would require 1.5 processors in the Virtual I/O Server to handle this workload. Such a workload could then be handled with either a 2-processor dedicated logical partition or a 1.5-processor shared-processor logical partition.

The following tables show the machine cycles per byte for a TCP-streaming workload.

Table 26. Shared Ethernet with threading option enabled

Type of Streaming	MTU 1500 rate and processor utilization	MTU 1500, cycles per byte	MTU 9000 rate and processor utilization	MTU 9000, cycles per byte
Simplex	112.8 MB at 80.6% processor	11.2	117.8 MB at 37.7% processor	5
Duplex	162.2 MB at 88.8% processor	8.6	217 MB at 52.5% processor	3.8

Table 27. Shared Ethernet with threading option disabled

Type of Streaming	MTU 1500 rate and processor utilization	MTU 1500, cycles per byte	MTU 9000 rate and processor utilization	MTU 9000, cycles per byte
Simplex	112.8 MB at 66.4% processor	9.3	117.8 MB at 26.7% processor	3.6
Duplex	161.6 MB at 76.4% processor	7.4	216.8 MB at 39.6% processor	2.9

The following tables show the machine cycles per transaction for a request and response workload. A transaction is defined as a round-trip request and reply size.

Table 28. Shared Ethernet with threading option enabled

Size of transaction	Transactions per second and Virtual I/O Server utilization	MTU 1500 or 9000, cycles per transaction
Small packets (64 bytes)	59,722 TPS at 83.4% processor	23,022
Large packets (1024 bytes)	51,956 TPS at 80% processor	25,406

Table 29. Shared Ethernet with threading option disabled

Size of transaction	Transactions per second and Virtual I/O Server utilization	MTU 1500 or 9000, cycles per transaction
Small packets (64 bytes)	60,249 TPS at 65.6% processor	17,956
Large packets (1024 bytes)	53,104 TPS at 65% processor	20,196

The preceding tables demonstrate that the threading option of the shared Ethernet adds approximately 16 – 20% more machine cycles per transaction for MTU 1500 streaming, and approximately 31% to 38% more machine cycles per transaction for MTU 9000. The threading option adds more machine cycles per transaction at lower workloads due to the threads being started for each packet. At higher workload rates, like full duplex or the request and response workloads, the threads can run longer without waiting and being redispatched. The thread option is a per-shared Ethernet option that can be configured by Virtual I/O Server commands. Disable the thread option if the shared Ethernet is running in a Virtual I/O Server logical partition by itself (without virtual SCSI in the same logical partition).

You can enable or disable threading using the **-attr thread** option of the mkvdev command. To enable threading, use the `-attr thread=1` option. To disable threading, use the `-attr thread=0` option. For example, the following command disables threading for Shared Ethernet Adapter `ent1`:

```
mkvdev -sea ent1 -vadapter ent5 -default ent5 -defaultid 1 -attr thread=0
```

Sizing a Virtual I/O Server for shared Ethernet on a shared processor logical partition

Creating a shared-processor logical partition for a Virtual I/O Server can be done if the Virtual I/O Server is running slower-speed networks (for example 10/100 Mb) and a full processor logical partition is not needed. It is recommended that this be done only if the Virtual I/O Server workload is less than half a processor or if the workload is inconsistent. Configuring the Virtual I/O Server logical partition as uncapped might also allow it to use more processor cycles as needed to handle inconsistent throughput. For example, if the network is used only when other processors are idle, the Virtual I/O Server logical partition might be able to use other machine cycles and could be created with minimal processor to handle light workload during the day but the uncapped processor could use more machine cycles at night.

If you are creating a Virtual I/O Server in a shared-processor logical partition, add additional processor entitlement as a sizing contingency.

Memory allocation:

Find information about memory allocation and sizing.

In general, 512 MB of memory per logical partition is sufficient for most configurations. Enough memory must be allocated for the Virtual I/O Server data structures. Ethernet adapters and virtual devices use dedicated receive buffers. These buffers are used to store the incoming packets, which are then sent over the outgoing device.

A physical Ethernet adapter typically uses 4 MB for MTU 1500 or 16 MB for MTU 9000 for dedicated receive buffers for gigabit Ethernet. Other Ethernet adapters are similar. Virtual Ethernet, typically uses 6 MB for dedicated receive buffers. However, this number can vary based on workload. Each instance of a physical or virtual Ethernet would need memory for this number of buffers. In addition, the system has an mbuf buffer pool per processor that is used if additional buffers are needed. These mbufs typically occupy 40 MB.

Configuration requirements for shared memory

Review the requirements for the system, Virtual I/O Server (VIOS), logical partitions, and paging space devices so that you can successfully configure shared memory.

System requirements

- The server must be a POWER6 processor-based server.
- The server firmware must be at release 3.4.2, or later.
- The Hardware Management Console (HMC) must be at version 7 release 3.4.2, or later.
- The Integrated Virtualization Manager must be at version 2.1.1, or later.
- The PowerVM Active Memory Sharing technology must be activated. The PowerVM Active Memory Sharing technology is available with the PowerVM Enterprise Edition for which you must obtain and enter a PowerVM Editions activation code.

Paging VIOS partition requirements

- Virtual I/O Server logical partitions that provide access to the paging space devices for the shared memory partitions that are assigned to the shared memory pool (hereafter referred to as *paging VIOS partitions*) cannot use shared memory. Paging VIOS partitions must use dedicated memory.
- Paging VIOS partitions must be at version 2.1.1, or later.
- On IVM-managed systems, all logical partitions that use shared memory (hereafter referred to as *shared memory partitions*) must use virtual resources provided by the management partition.
- On HMC-managed systems, consider configuring separate Virtual I/O Server logical partitions as server partitions and paging VIOS partitions. For example, configure one Virtual I/O Server logical partition to provide virtual resources to the shared memory partitions and configure another Virtual I/O Server logical partition as a paging VIOS partition.

- On HMC-managed systems, you can configure multiple Virtual I/O Server logical partitions to provide access to paging space devices. However, you can only assign up to two of those Virtual I/O Server logical partitions to the shared memory pool at any given time.

Requirements for shared memory partitions

- Shared memory partitions must use shared processors.
- You can assign only virtual adapters to shared memory partitions. This means that you can dynamically add only virtual adapters to shared memory partitions. More specifically, you can assign only virtual SCSI client adapters, virtual Ethernet adapters, virtual fibre channel client adapters, and virtual serial adapters to AIX and Linux shared memory partitions. You can assign only virtual SCSI client adapters, virtual Ethernet adapters, and virtual serial server adapters to IBM i shared memory partitions. You cannot assign Host Ethernet Adapters (HEA) or host connection adapters (HCA) to shared memory partitions.
- Shared memory partitions cannot use the barrier synchronization register.
- Shared memory partitions cannot use huge pages.
- AIX must be at version 6.1 Technology Level 3, or later, to run in a shared memory partition.
- IBM i must be at version 6 release 1 with PTF SI32798 to run in a shared memory partition.
- SUSE Linux Enterprise Server must be at version 11, or later, to run in a shared memory partition.
- You cannot configure IBM i logical partitions that provide virtual resources to other logical partitions as shared memory partitions. Logical partitions that provide virtual resources to other logical partitions in a shared memory environment must be Virtual I/O Server logical partitions.

Requirements for paging space devices

- The paging space devices for an AIX or Linux shared memory partitions must be at least the size of the maximum logical memory of the shared memory partition.
- The paging space devices for IBM i shared memory partitions must be at least the size of the maximum logical memory of the shared memory partition plus 8 KB for every megabyte. For example, if the maximum logical memory of the shared memory partition is 16 GB, its paging space device must be at least 16.125 GB.
- Paging space devices can only be assigned to one shared memory pool at a time. You cannot assign the same paging space device to a shared memory pool on one system and to another shared memory pool on another system at the same time.
- Paging space devices that are accessed by a single paging VIOS partition must meet the following requirements:
  - They can be physical or logical volumes.
  - They can be located in physical storage on the server or on a storage area network (SAN).
- Paging space devices that are accessed redundantly by two paging VIOS partitions must meet the following requirements:
  - They must be physical volumes.
  - They must be located on a SAN.
  - They must be configured with global IDs.
  - They must be accessible to both paging VIOS partitions.
  - The reserve attribute must be set to no reserve. (The Virtual I/O Server automatically sets the reserve attribute to no reserve when you add the paging space device to the shared memory pool.)
- Physical volumes that are configured as paging space devices cannot belong to a volume group, such as the rootvg volume group.
- Logical volumes that are configured as paging space devices must be located in a volume group that is dedicated for paging space devices.

- Paging space devices must be available. You cannot use the physical volume or logical volume as a paging space device if it is already configured as a paging space device or virtual disk for another logical partition.
- Paging space devices cannot be used to boot a logical partition.
- After you assign a paging space device to the shared memory pool, you must manage the device by using the Create/Modify Shared Memory Pool wizard on the HMC or the View/Modify Shared Memory Pool page on the Integrated Virtualization Manager. Do not change or remove the device using other management tools.

Redundancy considerations

Redundancy options are available at several levels in the virtual I/O environment. Multipathing, mirroring, and RAID redundancy options exist for the Virtual I/O Server and some client logical partitions. Ethernet Link Aggregation (also called EtherChannel) is also an option for the client logical partitions, and the Virtual I/O Server provides Shared Ethernet Adapter failover. There is also support for node failover (HACMP™) for nodes using virtual I/O resources.

This section contains information about redundancy for both the client logical partitions and the Virtual I/O Server. While these configurations help protect from the failure of one of the physical components, such as a disk or network adapter, they might cause the client logical partition to lose access to its devices if the Virtual I/O Server fails. The Virtual I/O Server can be made redundant by running a second instance of it in another logical partition. When running two instances of the Virtual I/O Server, you can use LVM mirroring, multipath I/O, network interface backup, or multipath routing with dead gateway detection in the client logical partition to provide highly available access to virtual resources hosted in separate Virtual I/O Server logical partitions.

Client logical partitions

This topic includes redundancy considerations for client logical partitions. MPIO, HACMP, and mirroring for the client logical partition are discussed.

Multipath I/O:

View Multipath I/O (MPIO) information for client logical partitions.

Multiple virtual SCSI or virtual fibre channel adapters in a client logical partition can access the same disk through multiple Virtual I/O Server logical partitions. This section describes a virtual SCSI multipath device configuration. If correctly configured, the client recognizes the disk as a multipath device. If you are using PowerVM Active Memory Sharing technology (or shared memory), you can also use a multipath configuration to allow two paging VIOS logical partitions to access common paging space devices.

MPIO is not available for IBM i client logical partitions. Instead, you must use mirroring to create redundancy. For more information, see “Mirroring for client logical partitions” on page 71.

Not all virtual SCSI devices are capable of MPIO. To create an MPIO configuration, the exported device at the Virtual I/O Server must conform to the following rules:

- The device must be backed by a physical volume. Logical volume-backed virtual SCSI devices are not supported in an MPIO configuration.
- The device must be accessible from multiple Virtual I/O Server logical partitions.
- The device must be an MPIO-capable device.

  Note: MPIO-capable devices are those that contain a unique identifier (UDID) or IEEE volume identifier. For instructions about how to determine whether disks have a UDID or IEEE volume identifier, see “Identifying exportable disks” on page 102.

When setting up an MPIO configuration for virtual SCSI devices on the client logical partition, you must consider the reservation policy of the device on the Virtual I/O Server. To use an MPIO configuration at the client, none of the virtual SCSI devices on the Virtual I/O Server can be reserving the virtual SCSI device. Ensure the `reserve_policy` attribute of the device is set to `no_reserve`.

Failover is the only supported behavior for MPIO virtual SCSI disks on the client logical partition.

Related tasks

"Setting the reserve policy attributes of a device" on page 99
In some configurations, you must consider the reservation policy of the device on the Virtual I/O Server.

"Scenario: Configuring Multi-Path I/O for AIX client logical partitions" on page 49
Multi-Path I/O (MPIO) helps provide increased availability of virtual SCSI resources by providing redundant paths to the resource. This topic describes how to set up Multi-Path I/O for AIX client logical partitions.

Related reference

"Configuration requirements for shared memory" on page 68
Review the requirements for the system, Virtual I/O Server (VIOS), logical partitions, and paging space devices so that you can successfully configure shared memory.

Mirroring for client logical partitions:

Achieve mirroring for client logical partitions by using two virtual SCSI adapters.

The client partition can mirror its logical volumes using two virtual SCSI client adapters. Each of these adapters should be assigned to separate Virtual I/O Server partitions. The two physical disks are each attached to a separate Virtual I/O Server partition and made available to the client partition through a virtual SCSI server adapter. This configuration protects virtual disks in a client partition against the failure of any of the following:

- One physical disk
- One physical adapter
- One Virtual I/O Server

The performance of your system might be impacted when using a RAID 1 configuration.

High Availability Cluster Multi-Processing:

Learn about High Availability Cluster Multi-Processing (HACMP) in the Virtual I/O Server.

HACMP supports certain configurations that utilize the Virtual I/O Server, virtual SCSI and virtual networking capabilities. For the most recent support and configuration information, see the HACMP for System p Web site. For HACMP documentation, see High Availability Cluster Multi-Processing for AIX.

For IBM i client partitions, you must use mirroring to create redundancy. For details, see "Mirroring for client logical partitions."

HACMP and virtual SCSI

Be aware of the following considerations when implementing HACMP and virtual SCSI:

- The volume group must be defined as Enhanced Concurrent Mode. Enhanced Concurrent Mode is the preferred mode for sharing volume groups in HACMP clusters because volumes are accessible by multiple HACMP nodes. If file systems are used on the standby nodes, those file systems are not mounted until the point of failover. If shared volumes are accessed directly (without file systems) in Enhanced Concurrent Mode, these volumes are accessible from multiple nodes, and as a result, access must be controlled at a higher layer.

- If any one cluster node accesses shared volumes through virtual SCSI, then all nodes must. This means that disks cannot be shared between a logical partition using virtual SCSI and a node directly accessing those disks.
- All volume group configuration and maintenance on these shared disks is done from the HACMP nodes, not from the Virtual I/O Server.

HACMP and virtual Ethernet

Be aware of the following considerations when implementing HACMP and virtual Ethernet:

- IP Address Takeover (IPAT) by way of aliasing must be used. IPAT by way of Replacement and MAC Address Takeover are not supported.
- Avoid using the HACMP PCI Hot Plug facility in a Virtual I/O Server environment. PCI Hot Plug operations are available through the Virtual I/O Server. When an HACMP node is using virtual I/O, the HACMP PCI Hot Plug facility is not meaningful because the I/O adapters are virtual rather than physical.
- All virtual Ethernet interfaces defined to HACMP should be treated as single-adapter networks. In particular, you must use the **ping_client_list** attribute to monitor and detect failure of the network interfaces.
- If the Virtual I/O Server has multiple physical interfaces on the same network, or if there are two or more HACMP nodes using the Virtual I/O Server in the same frame, HACMP is not informed of, and does not react to, single physical interface failures. This does not limit the availability of the entire cluster because the Virtual I/O Server routes traffic around the failure.
- If the Virtual I/O Server has only a single physical interface on a network, failure of that physical interface is detected by HACMP. However, that failure isolates the node from the network.

Link Aggregation or EtherChannel devices:

A Link Aggregation, or EtherChannel, device is a network port-aggregation technology that allows several Ethernet adapters to be aggregated. The adapters can then act as a single Ethernet device. Link Aggregation helps provide more throughput over a single IP address than would be possible with a single Ethernet adapter.

For example, `ent0` and `ent1` can be aggregated to `ent3`. The system considers these aggregated adapters as one adapter, and all adapters in the Link Aggregation device are given the same hardware address, so they are treated by remote systems as if they are one adapter.

Link Aggregation can help provide more redundancy because individual links might fail, and the Link Aggregation device will fail over to another adapter in the device to maintain connectivity. For example, in the previous example, if `ent0` fails, the packets are automatically sent on the next available adapter, `ent1`, without disruption to existing user connections. `ent0` automatically returns to service on the Link Aggregation device when it recovers.

You can configure a Shared Ethernet Adapter to use a Link Aggregation, or EtherChannel, device as the physical adapter.

Shared Ethernet Adapter failover:

Shared Ethernet Adapter failover provides redundancy by configuring a backup Shared Ethernet Adapter on a different Virtual I/O Server logical partition that can be used if the primary Shared Ethernet Adapter fails. The network connectivity in the client logical partitions continues without disruption.

A Shared Ethernet Adapter is comprised of a physical adapter (or several physical adapters grouped under a Link Aggregation device) and one or more virtual Ethernet adapters. It can provide layer 2 connectivity to multiple client logical partitions through the virtual Ethernet adapters.

The Shared Ethernet Adapter failover configuration uses the priority value given to the virtual Ethernet adapters during their creation to determine which Shared Ethernet Adapter will serve as the primary and which will serve as the backup. The Shared Ethernet Adapter that has the virtual Ethernet configured with the numerically lower priority value will be used preferentially as the primary adapter. For the purpose of communicating between themselves to determine when a failover should take place, Shared Ethernet Adapters in failover mode use a VLAN dedicated for such traffic, called the *control channel*. For this reason, a virtual Ethernet (created with a PVID that is unique on the system) must be specified as the control channel virtual Ethernet when each Shared Ethernet Adapter is created in failover mode. Using the control channel, the backup Shared Ethernet Adapter is notified when the primary adapter fails, and network traffic from the client logical partitions is sent over the backup adapter. If and when the primary Shared Ethernet Adapter recovers from its failure, it again begins actively bridging all network traffic.

A Shared Ethernet Adapter in failover mode might optionally have more than one trunk virtual Ethernet. In this case, all the virtual Ethernet adapters in a Shared Ethernet Adapter must have the same priority value. Also, the virtual Ethernet adapter used specifically for the control channel does not need to have the trunk adapter setting enabled. The virtual Ethernet adapters used for the control channel on each Shared Ethernet Adapter in failover mode must have an identical PVID value, and that PVID value must be unique in the system, so that no other virtual Ethernet adapters on the same system are using that PVID.

To ensure prompt recovery times, when you enable the Spanning Tree Protocol on the switch ports connected to the physical adapters of the Shared Ethernet Adapter, you can also enable the portfast option on those ports. The portfast option allows the switch to immediately forward packets on the port without first completing the Spanning Tree Protocol. (Spanning Tree Protocol blocks the port completely until it is finished.)

The Shared Ethernet Adapter is designed to prevent network loops. However, as an additional precaution, you can enable Bridge Protocol Data Unit (BPDU) Guard on the switch ports connected to the physical adapters of the Shared Ethernet Adapter. BPDU Guard detects looped Spanning Tree Protocol BPDU packets and shuts down the port. This helps prevent broadcast storms on the network. A *broadcast storm* is a situation where one message that is broadcast across a network results in multiple responses. Each response generates more responses, causing excessive transmission of broadcast messages. Severe broadcast storms can block all other network traffic, but they can usually be prevented by carefully configuring a network to block illegal broadcast messages.

Note: When the Shared Ethernet Adapter is using GARP VLAN Registration Protocol (GVRP), it generates BPDU packets, which causes BPDU Guard to shut down the port unnecessarily. Therefore, when the Shared Ethernet Adapter is using GVRP, do not enable BPDU Guard.

For information about how to enable the Spanning Tree Protocol, the portfast option, and BPDU Guard on the ports, see the documentation provided with the switch.

Related tasks

"Scenario: Configuring Shared Ethernet Adapter failover" on page 44
Use this article to help you become familiar with typical Shared Ethernet Adapter failover scenario.

Virtual I/O Server logical partition

Redundancy options for the Virtual I/O Server include multi-pathing, Redundant Array of Independent Disks (RAID) configurations, and Link Aggregation (or EtherChannel).

Multipathing:

Multipathing for the physical storage within the Virtual I/O Server provides failover physical path redundancy and load-balancing. The multipathing solutions available in the Virtual I/O Server include MPIO as well as solutions provided by the storage vendors.

For information about supported storage and multipathing software solutions, see the datasheet available on the Virtual I/O Server Support for UNIX servers and Midrange servers Web site.

RAID:

Redundant Array of Independent Disks (RAID) solutions provide for device level redundancy within the Virtual I/O Server. Some RAID options, such as LVM mirroring and striping, are provided by the Virtual I/O Server software, while other RAID options are made available by the physical storage subsystem.

See the Virtual I/O Server datasheet available on the Virtual I/O Server Support for UNIX servers and Midrange servers Web site for supported hardware RAID solutions.

Link Aggregation or EtherChannel devices:

A Link Aggregation, or EtherChannel, device is a network port-aggregation technology that allows several Ethernet adapters to be aggregated. The adapters can then act as a single Ethernet device. Link Aggregation helps provide more throughput over a single IP address than would be possible with a single Ethernet adapter.

For example, `ent0` and `ent1` can be aggregated to `ent3`. The system considers these aggregated adapters as one adapter, and all adapters in the Link Aggregation device are given the same hardware address, so they are treated by remote systems as if they are one adapter.

Link Aggregation can help provide more redundancy because individual links might fail, and the Link Aggregation device will fail over to another adapter in the device to maintain connectivity. For example, in the previous example, if `ent0` fails, the packets are automatically sent on the next available adapter, `ent1`, without disruption to existing user connections. `ent0` automatically returns to service on the Link Aggregation device when it recovers.

You can configure a Shared Ethernet Adapter to use a Link Aggregation, or EtherChannel, device as the physical adapter.

Redundancy configuration using virtual fibre channel adapters

Redundancy configurations help protect your network from physical adapter failures as well as Virtual I/O Server failures.

With N_Port ID Virtualization (NPIV), you can configure the managed system so that multiple logical partitions can access independent physical storage through the same physical fibre channel adapter. Each virtual fibre channel adapter is identified by a unique worldwide port name (WWPN), which means that you can connect each virtual fibre channel adapter to independent physical storage on a SAN.

Similar to virtual SCSI redundancy, virtual fibre channel redundancy can be achieved using Multi-path I/O (MPIO) and mirroring at the client partition. The difference between traditional redundancy with SCSI adapters and the NPIV technology using virtual fibre channel adapters, is that the redundancy occurs on the client, because only the client recognizes the disk. The Virtual I/O Server is essentially just a pipe. The second example below uses multiple Virtual I/O Server logical partitions to add redundancy at the Virtual I/O Server level as well.

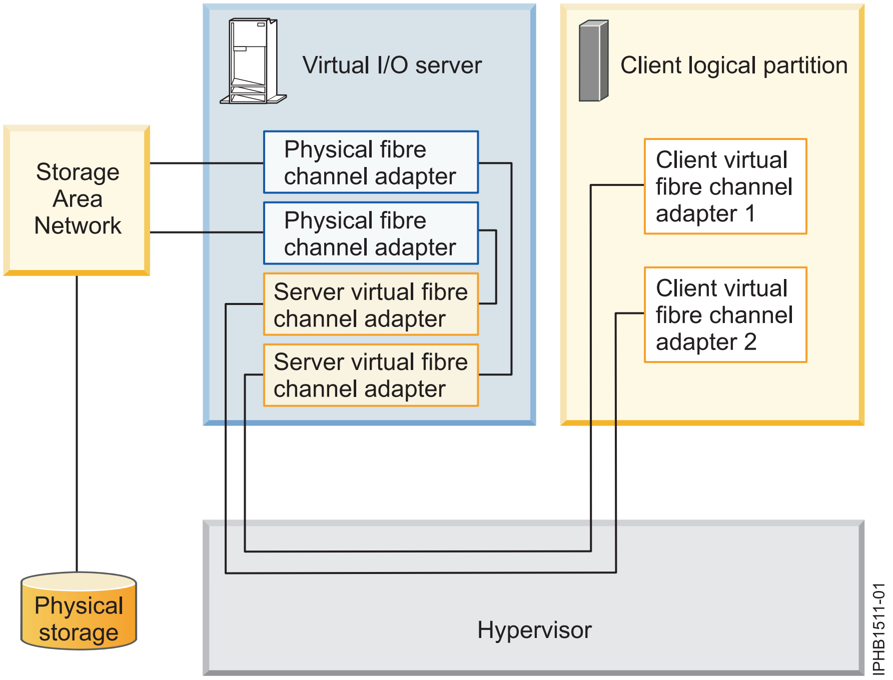
Example: Host bus adapter failover

This example uses Host bus adapter (HBA) failover to provide a basic level of redundancy for the client logical partition. The figure shows the following connections:

- The storage area network (SAN) connects physical storage to two physical fibre channel adapters located on the managed system.
- The physical fibre channel adapters are assigned to the Virtual I/O Server and support NPIV.

- The physical fibre channel ports are each connected to a virtual fibre channel adapter on the Virtual I/O Server. The two virtual fibre channel adapters on the Virtual I/O Server are connected to ports on two different physical fibre channel adapters in order to provide redundancy for the physical adapters.
- Each virtual fibre channel adapter on the Virtual I/O Server is connected to one virtual fibre channel adapter on a client logical partition. Each virtual fibre channel adapter on each client logical partition receives a pair of unique WWPNs. The client logical partition uses one WWPN to log into the SAN at any given time. The other WWPN is used when you move the client logical partition to another managed system.

The virtual fibre channel adapters always has a one-to-one relationship between the client logical partitions and the virtual fibre channel adapters on the Virtual I/O Server logical partition. That is, each virtual fibre channel adapter that is assigned to a client logical partition must connect to only one virtual fibre channel adapter on the Virtual I/O Server, and each virtual fibre channel on the Virtual I/O Server must connect to only one virtual fibre channel adapter on a client logical partition.


Virtual I/O server
Client logical partition
Storage Area Network
Physical fibre channel adapter
Physical fibre channel adapter
Server virtual fibre channel adapter
Server virtual fibre channel adapter
Client virtual fibre channel adapter 1
Client virtual fibre channel adapter 2
Physical storage
Hypervisor
IPHB1511-01

The client can write to the physical storage through client virtual fibre channel adapter 1 or 2. If a physical fibre channel adapter fails, the client uses the alternative path. This example does not show redundancy in the physical storage, but rather assumes it would be built into the SAN.

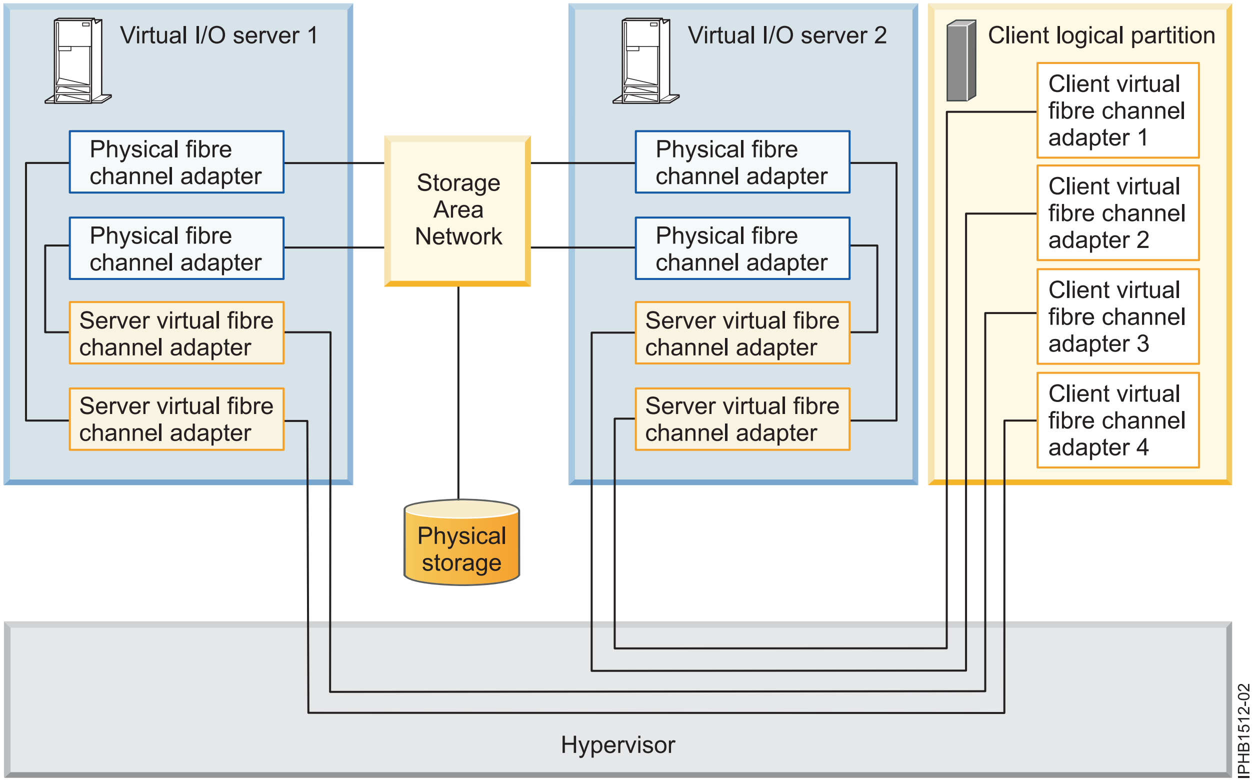
Note: It is recommended that you configure virtual fibre channel adapters from multiple logical partitions to the same HBA, or you configure virtual fibre channel adapters from the same logical partition to different HBAs.

Example: HBA and Virtual I/O Server failover

This example uses HBA and Virtual I/O Server failover to provide a more advanced level of redundancy for the client logical partition. The figure shows the following connections:

- The storage area network (SAN) connects physical storage to two physical fibre channel adapters located on the managed system.
- There are two Virtual I/O Server logical partitions to provide redundancy at the Virtual I/O Server level.
- The physical fibre channel adapters are assigned to their respective Virtual I/O Server and support NPIV.

- The physical fibre channel ports are each connected to a virtual fibre channel adapter on the Virtual I/O Server. The two virtual fibre channel adapters on the Virtual I/O Server are connected to ports on two different physical fibre channel adapters in order to provide redundancy for the physical adapters. A single adapter could have multiple ports.
- Each virtual fibre channel adapter on the Virtual I/O Server is connected to one virtual fibre channel adapter on a client logical partition. Each virtual fibre channel adapter on each client logical partition receives a pair of unique WWPNs. The client logical partition uses one WWPN to log into the SAN at any given time. The other WWPN is used when you move the client logical partition to another managed system.


Virtual I/O server 1
Physical fibre channel adapter
Physical fibre channel adapter
Server virtual fibre channel adapter
Server virtual fibre channel adapter
Storage Area Network
Physical storage
Virtual I/O server 2
Physical fibre channel adapter
Physical fibre channel adapter
Server virtual fibre channel adapter
Server virtual fibre channel adapter
Client logical partition
Client virtual fibre channel adapter 1
Client virtual fibre channel adapter 2
Client virtual fibre channel adapter 3
Client virtual fibre channel adapter 4
Hypervisor
IPHB1512-02

The client can write to the physical storage through virtual fibre channel adapter 1 or 2 on the client logical partition through VIOS 2. The client can also write to physical storage through virtual fibre channel adapter 3 or 4 on the client logical partition through VIOS 1. If a physical fibre channel adapter fails on VIOS 1, the client uses the other physical adapter connected to VIOS 1 or uses the paths connected through VIOS 2. If VIOS 1 fails, then the client uses the path through VIOS 2. This example does not show redundancy in the physical storage, but rather assumes it would be built into the SAN.

Considerations

These examples can become more complex as you add physical storage redundancy and multiple clients, but the concepts remain the same. Consider the following points:

- To avoid configuring the physical fibre channel adapter to be a single point of failure for the connection between the client logical partition and its physical storage on the SAN, do not connect two virtual fibre channel adapters from the same client logical partition to the same physical fibre channel adapter. Instead, connect each virtual fibre channel adapter to a different physical fibre channel adapter.
- Consider load balancing when mapping a virtual fibre channel adapter on the Virtual I/O Server to a physical port on the physical fiber channel adapter.

- Consider what level of redundancy already exists in the SAN to determine whether to configure multiple physical storage units.
- Consider using two Virtual I/O Server logical partitions. Since the Virtual I/O Server is central to communication between logical partitions and the external network, it is important to provide a level of redundancy for the Virtual I/O Server. Multiple Virtual I/O Server logical partitions require more resources as well, so you should plan accordingly.
- NPIV technology is useful when you want to move logical partitions between servers. For example, in active Partition Mobility, if you use the redundancy configurations above, in combination with physical adapters, you can stop all the I/O activity through the dedicated, physical adapter and direct all traffic through a virtual fibre channel adapter until the logical partition is successfully moved. The dedicated physical adapter would need to be connected to the same storage as the virtual path. Since you cannot migrate a physical adapter, all I/O activity is routed through the virtual path while you move the partition. After the logical partition is moved successfully, you need to set up the dedicated path (on the destination logical partition) if you want to use the same redundancy configuration you had configured on the original logical partition. Then the I/O activity can resume through the dedicated adapter, using the virtual fibre channel adapter as a secondary path.

Related information

Virtual I/O Server Deployment Examples

Configuring a virtual fibre channel adapter using the HMC

Configuring logical partitions to use virtual Fibre Channel on the Integrated Virtualization Manager

IBM PowerVM Live Partition Mobility

Security considerations

Review the security considerations for virtual SCSI, virtual Ethernet, and Shared Ethernet Adapter and the additional security options available.

IBM systems allow cross-partition device sharing and communication. Functions such as dynamic LPAR, shared processors, virtual networking, virtual storage, and workload management all require facilities to ensure that system-security requirements are met. Cross-partition and virtualization features are designed to not introduce any security exposure beyond what is implied by the function. For example, a virtual LAN connection would have the same security considerations as a physical network connection. Carefully consider how to utilize cross-partition virtualization features in high-security environments. Any visibility between logical partitions must be manually created through administrative system-configuration choices.

Using virtual SCSI, the Virtual I/O Server provides storage to client logical partitions. However, instead of SCSI or fiber cable, the connection for this functionality is done by the firmware. The virtual SCSI device drivers of the Virtual I/O Server and the firmware ensure that only the system administrator of the Virtual I/O Server has control over which logical partitions can access data on Virtual I/O Server storage devices. For example, a client logical partition that has access to a logical volume `lv001` exported by the Virtual I/O Server logical partition cannot access `lv002`, even if it is in the same volume group.

Similar to virtual SCSI, the firmware also provides the connection between logical partitions when using virtual Ethernet. The firmware provides the Ethernet switch functionality. The connection to the external network is provided by the Shared Ethernet Adapter function on the Virtual I/O Server. This part of the Virtual I/O Server acts as a layer-2 bridge to the physical adapters. A VLAN ID tag is inserted into every Ethernet frame. The Ethernet switch restricts the frames to the ports that are authorized to receive frames with that VLAN ID. Every port on an Ethernet switch can be configured to be a member of several VLANs. Only the network adapters, both virtual and physical, that are connected to a port (virtual or physical) that belongs to the same VLAN can receive the frames. The implementation of this VLAN standard ensures that the logical partitions cannot access restricted data.

Limitations and restrictions for IBM i client logical partitions

With Virtual I/O Server version 1.5 and later, you can install IBM i in a client logical partition. IBM i client logical partitions have unique system and storage requirements and considerations.

The following limitations and restrictions apply to IBM i client logical partitions of the Virtual I/O Server that are running on HMC-managed systems. IBM i client logical partitions that run on systems that are managed by the Integrated Virtualization Manager have additional limitations and restrictions. For details, see Limitations and restrictions for IBM i client partitions on systems managed by the Integrated Virtualization Manager.

Hardware and software prerequisites

- The managed system must be one of the following servers:
  - 9117-MMA
  - 9119-FHA
  - 9125-F2A
  - 9406-MMA
  - 9407-M15
  - 9408-M25
  - 9409-M50
- The Virtual I/O Server must be at version 1.5 or later.
- IBM i must be at 6.1 or later.

I/O, storage, and networking limitations

- The IBM i client logical partition can have up to 32 virtual SCSI devices under a single virtual adapter. It can have up to 16 disk units (logical volumes, physical volumes, or files) and up to 16 optical units.
- The maximum virtual disk size is 2 TB. If you are limited to 1 adapter and you have a storage requirement of 32 TB, for example, you might need to make your virtual disks the maximum size of 2 TB. However, in general, consider spreading the storage over multiple virtual disks with smaller capacities. This can help improve concurrency.
- Mirroring is the redundancy option for IBM i client logical partitions. However, you can use mutipathing and RAID for Virtual I/O Server redundancy.
- It is recommended that you assign the tape device to its own Virtual I/O Server adapter, as tape devices often send large amounts of data, which might affect the performance of any other device on the adapter.

Installing the Virtual I/O Server and client logical partitions

Find instructions for installing the Virtual I/O Server and client logical partitions by deploying a system plan or manually creating the logical partition and logical partition profiles and installing the Virtual I/O Server and client operating systems.

These instructions apply to installing the Virtual I/O Server and client logical partitions on a system that is managed by a Hardware Management Console (HMC). If you plan to install the Virtual I/O Server on a system that is not managed by an HMC, then you need to install the Integrated Virtualization Manager. For instructions, see Installing the Integrated Virtualization Manager.

The installation procedures vary depending on the following factors:

- The version of HMC attached to the managed system on which you plan to install the Virtual I/O Server and client logical partitions. HMC version 7 displays a different interface than prior versions of the HMC. HMC version 7 also provides the ability to deploy a system plan that includes the Virtual I/O Server and client logical partitions.

- Whether you plan to deploy a system plan that includes the Virtual I/O Server and client logical partitions. When you deploy a system plan, the HMC automatically performs the following tasks based on the information provided in the system plan:
  - Creates the Virtual I/O Server logical partition and logical partition profile.
  - Installs the Virtual I/O Server and provisions virtual resources.
  - Creates the client logical partitions and logical partition profiles.
  - Installs the AIX and Linux operating systems on client logical partitions. The HMC must be at V7R3.3.0, or later.

Related information

Installing the Virtual I/O Server using NIM

Installing the Virtual I/O Server and client logical partitions by deploying a system plan

When you deploy a system plan that includes the Virtual I/O Server and, optionally, client logical partitions, the Deploy System Plan wizard creates the Virtual I/O Server logical partition and the logical partition profile, and installs the Virtual I/O Server and client logical partitions.

Before you start, ensure that you meet the following requirements:

- The system to which you plan to deploy the system plan is managed by a Hardware Management Console (HMC).
- The HMC is at version 7 or later.
- If you plan to deploy different entities of the Virtual I/O Server configuration at different times, ensure that the HMC is at version V7R3.3.0, or later. (Virtual I/O Server entities include Shared Ethernet Adapters, EtherChannel adapters, or Link Aggregation devices, storage pools, and backing devices.) If the HMC is not at V7R3.3.0, or later, system plans that include the Virtual I/O Server can be deployed only to new systems, or to systems that do not already have a Virtual I/O Server logical partition configured. (The Virtual I/O Server can be installed, but not configured.) More specifically, no Virtual I/O Server entities can be configured on the managed system, including Shared Ethernet Adapters, EtherChannel adapters, or Link Aggregation devices, storage pools, and backing devices.
- If you plan to deploy a system plan that includes AIX or Linux installation information for at least one client logical partition, ensure that you meet the following requirements:
  - The HMC must be at V7R3.3.0.
  - The client logical partition does not have an operating system already installed. The HMC installs AIX and Linux on client logical partitions that do not already have an operating system installed. If the client logical partition already has an operating system installed, the HMC does not deploy the operating system specified in the system plan.

Entering the activation code for PowerVM Editions using the HMC version 7

Use these instructions to enter the PowerVM Editions (or Advanced POWER Virtualization) activation code using the Hardware Management Console (HMC) version 7, or later.

If PowerVM Editions is not enabled on your system, you can use the HMC to enter the activation code that you received when you ordered the feature.

Use the following procedure to enter the activation code for the PowerVM Standard Edition and the PowerVM Enterprise Edition. For information about the PowerVM Editions, see PowerVM Editions overview.

To enter your activation code, follow these steps:

1. In the Navigation area, expand **Systems Management**.
2. Select **Servers**.

3. In the contents area, select the managed system on which you plan to use PowerVM Editions. For example, this might be the system on which you plan to install the Virtual I/O Server, or it might be the system in which you plan to use the Micro-Partitioning™ technology.
4. Click **Tasks** and select **Capacity on Demand (CoD)** → **Advanced POWER Virtualization** → **Enter Activation Code**.
5. Enter your activation code and click **OK**.

Importing a system plan into an HMC

You can import a system-plan file into a Hardware Management Console (HMC) from various types of media, a remote FTP site, or the computer from which you remotely access the HMC. You can then deploy the imported system plan to a system that the HMC manages.

You can import a system-plan file into the HMC from any of the following locations:

- From the computer on which you remotely access the HMC.
- From various media that is mounted on the HMC, such as optical discs or USB drives.
- From a remote site by using FTP. To use this option, you must fulfill the following requirements:
  - The HMC must have a network connection to the remote site.
  - An FTP server must be active on the remote site.
  - Port 21 must be open on the remote site.

Note: You cannot import a system plan that has an identical name to any system plan that is available on the HMC.

To import a system-plan file, you must be a super administrator. For more information about user roles, see Managing HMC users and tasks.

To import a system-plan file into the HMC, complete the following steps:

1. In the navigation area of the HMC, select **System Plans**.
2. In the tasks area, select **Import System Plan**. The Import System Plan window opens.
3. Select the source of the system-plan file that you want to import. Use the following table to complete the appropriate steps for importing the system plan from the selected source location of the file.

Source of the system plan to import	Complete the following steps:
This computer	1. Select **Import from this computer to the HMC**. 2. Click **Import** to display the Upload File window. 3. Click **Browse**. 4. Select the system-plan file that you want to import and click **Open**. 5. Click **OK** to upload the file.

Source of the system plan to import	Complete the following steps:
Media	1. Select **Import from media**. 2. In the **System plan file name** field, enter the name of the system-plan file. **Note:** The name of the system-plan file must end with the .sysplan file name suffix and can use alphanumeric characters only. 3. In the **Sub-directory on media** field, enter the path in which the system-plan file is located on the media. **Note:** Specify the subdirectory location only, rather than the fully qualified path and file name. 4. Click **Import** to display the Select Media Device window. 5. Select the media that contains the system-plan file that you want to import. 6. Click **OK**.
Remote FTP site	1. Select **Import from a remote FTP site**. 2. In the **System plan file name** field, enter the name of the system-plan file. **Note:** The name of the system-plan file must end with the .sysplan file name suffix and can use alphanumeric characters only. 3. In the **Remote site hostname** field, enter the host name or IP address of the remote FTP site. 4. In the **User ID** field, enter the user ID to use to access the remote FTP site. 5. In the **Password** field, enter the password to use to access the remote FTP site. 6. In the **Remote directory** field, enter the path in which the system-plan file is located on the remote FTP site. If you do not enter a path, the HMC uses the default path specified on the remote FTP site.

4. Click **Import**. If the HMC returns an error, return to the **Import System Plan** window and verify that the information you entered is correct. If necessary, click **Cancel**, return to step 2, and redo the procedure, ensuring that the information you specify at each step is correct.

Note: As an alternative to the HMC Web user interface, you can use the `cpysysplan` command from the HMC command line interface to import a system plan.

When you complete the process of importing the system-plan file, you can deploy the system plan in the system-plan file to a system that the HMC manages. If you imported the system-plan file from media, you can unmount the media by using the umount command from the HMC command line interface.

Related tasks

"Deploying a system plan by using the HMC"
You can use the Hardware Management Console (HMC) to deploy all or part of a system plan to a managed system.

Related information

Managing the Hardware Management Console
This publication provides system administrators and system operators with information about using the Hardware Management Console.

Deploying a system plan by using the HMC

You can use the Hardware Management Console (HMC) to deploy all or part of a system plan to a managed system.

When you deploy a system plan, the HMC creates logical partitions on the managed system according to the specifications in the system plan. Depending on the contents of the system plan, you can also install operating environments on the logical partitions in the plan, including the Virtual I/O Server (VIOS), AIX or Linux.

Note: The HMC cannot install the IBM i operating environment on a logical partition.

If the plan contains VIOS provisioning information for a logical partition, such as storage assignments and virtual networking for the client logical partitions of the VIOS. the HMC can make these resource assignments for the client logical partitions.

You do not have to deploy a system plan in its entirety, but can instead partially deploy a system plan on the target system by selecting which logical partitions in the plan to deploy. You can run the Deploy System Plan Wizard again at another time to deploy the remainder of the logical partitions in the system plan. However, if you select a VIOS partition to be deployed, the wizard deploys all the VIOS provisioning items that are planned for that partition even if the client logical partition that uses the provisioned item is not selected for deployment.

If the system plan contains installation information for the VIOS, you can use the Deploy System Plan Wizard to install the VIOS and to set up virtual networking and storage resources for the client logical partitions of the VIOS.

Before you deploy a system plan, complete the following tasks:

- Ensure that the system-plan file exists on the HMC. If the system-plan file does not exist on the HMC, you must import the system-plan file into the HMC. For instructions, see Importing a system plan into an HMC.
- Ensure that you meet all the appropriate requirements for deploying the system plan. See System plan deployment requirements for more information.

Deploying a system plan

To use the HMC to deploy a system plan on a managed system, complete the following steps:

1. In the navigation area of the HMC, select **System Plans**.
2. In the contents area, select the system plan that you want to deploy.
3. Select **Tasks** → **Deploy system plan**. The Deploy System Plan Wizard starts.
4. On the Welcome page, complete the following steps:
   a. Select the system-plan file that contains the system plan that you want to deploy.

   b. Choose the managed system to which you want to deploy the system plan and click **Next**. If the system plan does not match the managed system to which you want to deploy the plan, the wizard displays a window that informs you of this. Click **OK** to continue or **Cancel** to select a different system plan.

      Note: If the system-plan file contains multiple system plans, the wizard provides a step so that you can select a specific system plan from the file. This step does not occur unless there is more than one system plan in the specified file.

5. On the Validation page, complete the following steps:
   a. Wait for the wizard to validate the managed system and its hardware against the system plan. The validation process can take several minutes.
   b. If the validation process completes successfully, click **Next**.
   c. If the validation process fails, correct the problems that the error messages describe, click **Cancel** to exit the wizard, and restart this procedure from the beginning. To help you correct any validation problems, you might want to create a system plan that is based on the current configuration of the managed system. Such a system plan allows you to compare the system plan that you want to deploy with the current configuration of the managed system. You can do this by using the Create System Plan task in the HMC, or you can run the following command from the HMC command line:

      ```
      mksysplan -m name_of_managed_system -f name_of_new_system_plan.sysplan
      ```

      This action creates a new system plan that you can view and compare to the old system plan to help diagnose any problems.

6. Optional: On the Partition Deployment page, if you do not want to create all of the logical partitions, partition profiles, virtual adapter types, or virtual adapters in the system plan, clear the boxes in the **Deploy** column beside the logical partitions, partition profiles, virtual adapter types, or virtual adapters that you do not want to create. Virtual serial adapters are required in virtual slots 0 and 1 for each logical partition. You cannot create the logical partition unless you create these virtual serial adapters.
7. Optional: On the Operating Environment Install page, if there is operating environment installation information specified in the system plan, complete the following steps:
   a. Select the operating environments that you want to deploy to the managed system for each logical partition. For HMC V7R3.2.0 or V7R3.1.0, you can deploy only the Virtual I/O Server operating environment. For HMC V7R3.3.0, or later, versions, you also can select to deploy the AIX or Linux operating environments if the system plan contains installation information for them.
   b. Enter the location of the Virtual I/O Server installation image.
   c. Enter or change late-binding installation settings for the specified Virtual I/O Server, AIX, or Linux operating environment. Late-binding installation settings are settings that are specific to the installation instance and must be supplied during the installation step to ensure that the settings are accurate for the installation instance. For example, you can enter the IP address of the target logical partition on which you are installing the operating environment.

      Note: If you need to use automatic installation files to deploy an operating environment, you cannot add them during the HMC deployment process. You must use the System Planning Tool (SPT) to create any necessary automatic installation files separately and attach them to the system plan prior to deploying the system plan.
   d. Save any changes that you make to late-binding installation settings. You can save them to the current system-plan file or to a new system-plan file.
8. On the Summary page, review the system deployment step order and click **Finish**. The HMC uses the system plan to create the specified logical partitions and to install any specified operating environments. This process can take several minutes.

After you finish the deployment of the system plan, install operating environments and software on the logical partitions, if they did not install as part of system plan deployment.

Related tasks

"Importing a system plan into an HMC" on page 80
You can import a system-plan file into a Hardware Management Console (HMC) from various types of media, a remote FTP site, or the computer from which you remotely access the HMC. You can then deploy the imported system plan to a system that the HMC manages.

Related information

Logical Partitioning
This publication describes how to use a Hardware Management Console (HMC) to create and maintain logical partitions.

Managing the Hardware Management Console
This publication provides system administrators and system operators with information about using the Hardware Management Console.

Finishing the Virtual I/O Server installation

After you install Virtual I/O Server, you must check for updates, set up remote connects, create additional user IDs, and so on.

This procedure assumes that Virtual I/O Server is installed. For instructions, see "Installing the Virtual I/O Server and client logical partitions" on page 78.

To finish the installation, complete the following steps:

1. Accept the software maintenance terms and conditions, and the Virtual I/O Server product license. For instructions, see "Viewing and accepting the Virtual I/O Server license" on page 88.
2. Check for updates to the Virtual I/O Server. For instructions, see "Updating the Virtual I/O Server" on page 126.
3. Set up remote connections to the Virtual I/O Server. For instructions, see "Connecting to the Virtual I/O Server using OpenSSH" on page 144.
4. Optional: Create the following additional user IDs. After the installation, the only active user ID is the prime administrator (padmin). You can create the following additional user IDs: system administrator, service representative, and development engineer. For information about creating user IDs, see "Managing users on the Virtual I/O Server" on page 149.
5. Configure the TCP/IP connection for the Virtual I/O Server using the mktcpip command. You must complete this task before you can perform any dynamic logical partitioning operations. Alternatively, you can use the configuration assistance menu to configure TCP/IP connections. You can access the configuration assistance menu by running the cfgassist command.

When you are finished, do one of the following tasks:

- If you installed the Virtual I/O Server, client logical partitions, and operating systems by completely deploying a system plan, your setup is complete. For information about how to manage the Virtual I/O Server, see "Managing the Virtual I/O Server" on page 117.
- If you installed the Virtual I/O Server manually using HMC version 6 or version 7, you need to configure the Virtual I/O Server, create client logical partitions, and install client operating systems. For information, see "Configuring the Virtual I/O Server" on page 93 and the Logical partitioning. To view the PDF file of Logical partitioning, approximately 1 MB in size, see http://publib.boulder.ibm.com/infocenter/systems/scope/hw/topic/iphat/iphat.pdf .

Installing the Virtual I/O Server manually using the HMC version 7

You can create the Virtual I/O Server logical partition and logical partition profile and install the Virtual I/O Server using the Hardware Management Console (HMC) version 7 or later.

Before you start, ensure that the following statements are true:

- The system on which you plan install the Virtual I/O Server is managed by a Hardware Management Console (HMC).
- The HMC is at version 7 or later. If the HMC is at a version 6 or earlier, then see Installing the Virtual I/O Server manually using the HMC version 6.

Entering the activation code for PowerVM Editions using the HMC version 7

Use these instructions to enter the PowerVM Editions (or Advanced POWER Virtualization) activation code using the Hardware Management Console (HMC) version 7, or later.

If PowerVM Editions is not enabled on your system, you can use the HMC to enter the activation code that you received when you ordered the feature.

Use the following procedure to enter the activation code for the PowerVM Standard Edition and the PowerVM Enterprise Edition. For information about the PowerVM Editions, see PowerVM Editions overview.

To enter your activation code, follow these steps:

1. In the Navigation area, expand **Systems Management**.
2. Select **Servers**.
3. In the contents area, select the managed system on which you plan to use PowerVM Editions. For example, this might be the system on which you plan to install the Virtual I/O Server, or it might be the system in which you plan to use the Micro-Partitioning technology.
4. Click **Tasks** and select **Capacity on Demand (CoD)** → **Advanced POWER Virtualization** → **Enter Activation Code**.
5. Enter your activation code and click **OK**.

Creating the Virtual I/O Server logical partition and partition profile using HMC version 7

You can use the Hardware Management Console (HMC) version 7 to create a logical partition and partition profile for the Virtual I/O Server.

Before you start, ensure that the following statements are true:

- You are a super administrator or an operator.
- The PowerVM Editions (or Advanced POWER Virtualization) feature is activated. For instructions, see "Entering the activation code for PowerVM Editions using the HMC version 7" on page 79.

The Virtual I/O Server requires a minimum of 16 GB of disk space.

To create a logical partition and a partition profile on your server using the HMC, follow these steps:

1. In the Navigation area, expand **Systems Management**.
2. Select **Servers**.
3. In the contents area, select the server on which you want to create the partition profile.
4. Click **Tasks** and select **Configuration** → **Create Logical Partition** → **VIO Server**.
5. On the Create Partition page, enter a name and ID for the Virtual I/O Server partition.
6. On the Partition Profile page, complete the following steps:
   a. Enter a profile name for the Virtual I/O Server partition.
   b. Make sure that the **Use all the resources in the system** check box is cleared (not checked).
7. On the Processors page, decide if you want to use shared or dedicated processors (based on your environment) by making the appropriate selection.

8. On the Processing Settings page, enter the appropriate amount of processing units and virtual processors that you want to assign to the Virtual I/O Server partition.
9. On the Memory page, select the appropriate amount of memory that you want to assign to the Virtual I/O Server partition. The required minimum is 512 MB.
10. On the I/O page, select the physical I/O resources that you want in the Virtual I/O Server partition.
11. On the Virtual Adapters page, create the appropriate adapters for your environment.
12. On the Logical Host Ethernet Adapter (LHEA) page, configure one or more LHEAs for the Virtual I/O Server partition. (Host Ethernet Adapter is sometimes referred to as Integrated Virtual Ethernet.)
13. On the Optional Settings page, complete the following steps:
    a. Decide if you want connection monitoring by making the appropriate selection.
    b. If you want the Virtual I/O Server to start when the managed system starts, select the **Automatically start with managed system** option.
    c. Decide if you want to enable redundant error path reporting by making the appropriate selection.
    d. Select the boot mode for the Virtual I/O Server partition. In most cases, the **Normal** boot mode is the appropriate selection.
14. Verify your selections in the Profile Summary window and click **Finish**.

After you create the partition and partition profile, you are ready to install the Virtual I/O Server. For instructions, see one of the following procedures:
- "Installing the Virtual I/O Server from the HMC"
- "Installing the Virtual I/O Server from CD or DVD" on page 87

Installing the Virtual I/O Server from the HMC

Find instructions for installing the Virtual I/O Server from the HMC by using the installios command.

Before you start, complete the following tasks:
1. Ensure that the following statements are true:
    - There is an HMC attached to the managed system.
    - The Virtual I/O Server logical partition and logical partition profile are created. For instructions, see "Creating the Virtual I/O Server logical partition and partition profile using HMC version 7" on page 85.
    - The Virtual I/O Server logical partition has at least one Ethernet adapter and a 16 GB disk assigned to it.
    - You have **hmcsuperadmin** authority.
2. Gather the following information:
    - Static IP address for the Virtual I/O Server
    - Subnet mask for the Virtual I/O Server
    - Default gateway for the Virtual I/O Server

To install the Virtual I/O Server, follow these steps:
1. Insert the Virtual I/O Server CD or DVD into the HMC.
2. If you are installing the Virtual I/O Server through the public network interface, continue to step 3. If you are installing the Virtual I/O Server through a private network interface, type the following from the HMC command line:

    ```
    export INSTALLIOS_PRIVATE_IF=interface
    ```

    where *interface* is the network interface through which the installation should take place.
3. From the HMC command line, type:

    ```
    installios
    ```

4. Follow the installation instructions according to the system prompts.

After you install the Virtual I/O Server, finish the installation by checking for updates, setting up remote connections, creating additional user IDs, and so on. For instructions, see “Finishing the Virtual I/O Server installation” on page 84.

Installing the Virtual I/O Server from CD or DVD

Find instructions for installing the Virtual I/O Server from a CD or DVD device that is attached to the Virtual I/O Server logical partition.

Before you start, ensure that the following statements are true:

- There is an HMC attached to the managed system.
- The Virtual I/O Server logical partition and logical partition profile are created. For instructions, see “Creating the Virtual I/O Server logical partition and partition profile using HMC version 7” on page 85.
- A CD or DVD optical device is assigned to the Virtual I/O Server logical partition.

To install the Virtual I/O Server from CD or DVD, follow these steps:

1. Activate the Virtual I/O Server logical partition using the HMC version 7 (or later) or HMC version 6 (or earlier):
   - Activate the Virtual I/O Server using the HMC version 7 or later:
     a. Insert the Virtual I/O Server CD or DVD into the Virtual I/O Server logical partition.
     b. In the HMC navigation area, expand **Systems Management** → **Servers**.
     c. Select the server on which the Virtual I/O Server logical partition is located.
     d. In the contents area, select the Virtual I/O Server logical partition.
     e. Click **Tasks** → **Operations** → **Activate**. The Activate Partition menu opens with a selection of logical partition profiles. Ensure the correct profile is highlighted.
     f. Select **Open a terminal window or console session** to open a virtual terminal (vterm) window.
     g. Click **(Advanced)** to open the advanced options menu.
     h. For the boot mode, select **SMS**.
     i. Click **OK** to close the advanced options menu.
     j. Click **OK**. A virtual terminal window opens for the logical partition.
   - Activate the Virtual I/O Server using the HMC version 6 or earlier:
     a. Insert the Virtual I/O Server CD or DVD into the Virtual I/O Server logical partition.
     b. On the HMC, right-click the logical partition to open the menu.
     c. Click **Activate**. The Activate Partition menu opens with a selection of logical partition profiles. Ensure the correct profile is highlighted.
     d. Select **Open a terminal window or console session** to open a virtual terminal (vterm) window.
     e. Click **(Advanced)** to open the advanced options menu.
     f. For the boot mode, select **SMS**.
     g. Click **OK** to close the advanced options menu.
     h. Click **OK**. A virtual terminal window opens for the logical partition.
2. Select the boot device:
   a. Select **Select Boot Options** and press Enter.
   b. Select **Select Install/Boot Device** and press Enter.
   c. Select **Select 1st Boot Device** and press Enter.
   d. Select **CD/DVD** and press Enter.
   e. Select the media type that corresponds to the optical device and press Enter.
   f. Select the device number that corresponds to the optical device and press Enter.

   g. Set the boot sequence to configure the first boot device. The optical device is now the first device in the Current Boot Sequence list.
   h. Exit the SMS menu by pressing the x key, and confirm that you want to exit SMS.
3. Install the Virtual I/O Server:
   a. Select the desired console and press Enter.
   b. Select a language for the BOS menus and press Enter.
   c. Select **Start Install Now with Default Settings** and press Enter.
   d. Select **Continue with Install**. The system will reboot after the installation is complete.

After you install the Virtual I/O Server, finish the installation by checking for updates, setting up remote connects, creating additional user IDs, and so on. For instructions, see "Finishing the Virtual I/O Server installation" on page 84.

Finishing the Virtual I/O Server installation

After you install Virtual I/O Server, you must check for updates, set up remote connects, create additional user IDs, and so on.

This procedure assumes that Virtual I/O Server is installed. For instructions, see "Installing the Virtual I/O Server and client logical partitions" on page 78.

To finish the installation, complete the following steps:

1. Accept the software maintenance terms and conditions, and the Virtual I/O Server product license. For instructions, see "Viewing and accepting the Virtual I/O Server license."
2. Check for updates to the Virtual I/O Server. For instructions, see "Updating the Virtual I/O Server" on page 126.
3. Set up remote connections to the Virtual I/O Server. For instructions, see "Connecting to the Virtual I/O Server using OpenSSH" on page 144.
4. Optional: Create the following additional user IDs. After the installation, the only active user ID is the prime administrator (padmin). You can create the following additional user IDs: system administrator, service representative, and development engineer. For information about creating user IDs, see "Managing users on the Virtual I/O Server" on page 149.
5. Configure the TCP/IP connection for the Virtual I/O Server using the mktcpip command. You must complete this task before you can perform any dynamic logical partitioning operations. Alternatively, you can use the configuration assistance menu to configure TCP/IP connections. You can access the configuration assistance menu by running the cfgassist command.

When you are finished, do one of the following tasks:

- If you installed the Virtual I/O Server, client logical partitions, and operating systems by completely deploying a system plan, your setup is complete. For information about how to manage the Virtual I/O Server, see "Managing the Virtual I/O Server" on page 117.
- If you installed the Virtual I/O Server manually using HMC version 6 or version 7, you need to configure the Virtual I/O Server, create client logical partitions, and install client operating systems. For information, see "Configuring the Virtual I/O Server" on page 93 and the Logical partitioning. To view the PDF file of Logical partitioning, approximately 1 MB in size, see

  http://publib.boulder.ibm.com/infocenter/systems/scope/hw/topic/iphat/iphat.pdf .

Viewing and accepting the Virtual I/O Server license:

You must view and accept the license before using the Virtual I/O Server.

Before you start, ensure that the Virtual I/O Server logical partition profile is created and the Virtual I/O Server is installed. For instructions, see “Installing the Virtual I/O Server and client logical partitions” on page 78.

To view and accept the Virtual I/O Server license, complete the following steps:

1. Log in to the Virtual I/O Server using the **padmin** user ID.
2. Choose a new password. The software maintenance terms and conditions appear.
3. If Virtual I/O Server is at version 1.5 or later, view and accept the software maintenance terms and conditions.
   a. To view the software maintenance terms and conditions, type `v` on the command line and press **Enter**.
   b. To accept the software maintenance terms and conditions, type `a` on the command line and press **Enter**.
4. View and accept the Virtual I/O Server product license.

   Note: If you installed the Virtual I/O Server by deploying a system plan, then you have already accepted the Virtual I/O Server product license and do not need to complete this step.

   a. To view the Virtual I/O Server product license, type `license -ls` on the command line. By default, the license is displayed in English. To change the language in which the license is displayed, follow these steps:
      1) View the list of available locales to display the license by typing the following command:
      ```
      license -ls
      ```
      2) View the license in another language by typing the following command:
      ```
      license -view -lang Name
      ```
      For example, to view the license in Japanese, type the following command:
      ```
      license -view -lang ja_JP
      ```
   b. To accept the Virtual I/O Server product license, type `license -accept` on the command line.
5. In the installation program, English is the default language. If you need to change the language setting for the system, follow these steps:
   a. View the available languages by typing the following command:
   ```
   chlang -ls
   ```
   b. Change the language by typing the following command, replacing *Name* with the name of the language you are switching to:
   ```
   chlang -lang Name
   ```
   Note: If the language fileset is not installed, use the **-dev *Media*** flag to install it.
   For example, to install and change the language to Japanese, type the following command:
   ```
   chlang -lang ja_JP -dev /dev/cd0
   ```

Reinstalling the Virtual I/O Server of a paging VIOS partition

When you reinstall the Virtual I/O Server (VIOS) that is assigned to the shared memory pool (hereafter referred to as a *paging VIOS partition*), you need to reconfigure the shared memory environment. For example, you might need to add the paging space devices again to the shared memory pool.

The paging VIOS partitions store information about the paging space devices that are assigned to a shared memory pool. The Hardware Management Console (HMC) obtains information about the paging space devices that are assigned to the shared memory pool from the paging VIOS partitions. When you reinstall the VIOS, the information about the paging space devices is lost. For the paging VIOS partitions to regain the information, you must assign the paging space devices again to the share memory pool after you reinstall the VIOS.

The following table shows the reconfiguration tasks that you must perform in the shared memory environment when you resinstall the Virtual I/O Server of a paging VIOS partition.

Table 30. Shared memory reconfiguration tasks for reinstalling the Virtual I/O Server of a paging VIOS partition

Number of paging VIOS partitions that are assigned to the shared memory pool	Number of paging VIOS partitions for which you want to reinstall the VIOS	Reconfiguration steps	Instructions
1	1	1. Shut down all logical partitions that use shared memory (hereafter referred to as *shared memory partitions*). 2. Reinstall the VIOS. 3. Add the paging space devices again to the shared memory pool.	1. Shutting down and restarting logical partitions 2. Installing the Virtual I/O Server manually using the HMC version 7 3. Adding and removing paging space devices to and from the shared memory pool
2	1	1. Shut down each shared memory partition that uses the paging VIOS partition (that you plan to reinstall) as the primary or secondary paging VIOS partition. 2. Remove the paging VIOS partition from the shared memory pool. 3. Reinstall the VIOS. 4. Add the paging VIOS partition again to the shared memory pool.	1. Shutting down and restarting logical partitions 2. Removing a paging VIOS partition from the shared memory pool 3. Installing the Virtual I/O Server manually using the HMC version 7 4. Adding a paging VIOS partition to the shared memory pool
2	2	1. Shut down all the shared memory partitions. 2. Reinstall the VIOS of each paging VIOS partition. 3. Add the paging space devices again to the shared memory pool.	1. Shutting down and restarting logical partitions 2. Installing the Virtual I/O Server manually using the HMC version 7 3. Adding and removing paging space devices to and from the shared memory pool

Migrating the Virtual I/O Server

You can migrate the Virtual I/O Server logical partition from the Hardware Management Console (HMC) version 7, or later, or from a DVD device that is attached to the Virtual I/O Server logical partition.

Before you start, verify that the following statements are true:

- The system on which you plan to migrate the Virtual I/O Server is managed by a Hardware Management Console (HMC) version 7, or later.
- The Virtual I/O Server is at version 1.3, or later.
- The rootvg volume group has been assigned to the Virtual I/O Server.

Note: If you are using an Integrated Virtualization Manager (IVM) environment, see Migrating the Virtual I/O Server from DVD when using the Integrated Virtualization Manager.

In most cases, user configuration files from the previous version of the Virtual I/O Server are saved when the new version is installed. If you have two or more Virtual I/O Server logical partitions in your environment for redundancy, you are able to shut down and migrate one Virtual I/O Server logical partition without interrupting any clients. After the migration is complete and the Virtual I/O Server logical partition is running again, the logical partition will be available to clients without additional configuration.

Attention: Do not use the Virtual I/O Server **updateios** command to migrate the Virtual I/O Server.

Related information

Migrating the Virtual I/O Server using NIM

Migrating the Virtual I/O Server from the HMC

Find instructions for migrating the Virtual I/O Server from the Hardware Management Console (HMC) by using the installios command.

Before you start, verify the following statements are true:

- HMC is attached to the managed system.
- The Virtual I/O Server logical partition has at least one Ethernet adapter and a 16 GB disk assigned to it.
- You have **hmcsuperadmin** authority.
- You have the Virtual I/O Server migration media.

  Note: The migration media is separate from the installation media.
- The Virtual I/O Server is currently at version 1.3 or later.
- The rootvg volume group has been assigned to the Virtual I/O Server
- Back up the mksysb image before migrating Virtual I/O Server. Run the backupios command and save the mksysb image to a safe location.

To migrate the Virtual I/O Server, follow these steps:

1. Insert the **Virtual I/O Server migration DVD** into the HMC.
2. If you are installing the Virtual I/O Server through the public network interface, continue to step 3. If you are installing the Virtual I/O Server through a private network interface, type the following command from the HMC command line:

   ```
   export INSTALLIOS_PRIVATE_IF=interface
   ```

   where *interface* is the network interface through which the installation should take place.
3. From the HMC command line, type:

   ```
   installios
   ```

   Attention: Do not use the Virtual I/O Server updateios command to migrate the Virtual I/O Server.
4. Follow the installation instructions according to the system prompts.

After the migration is complete, the Virtual I/O Server logical partition is restarted to its preserved configuration prior to the migration installation. It is recommended to perform the following tasks:

- Verify that migration was successful by checking results of the installp command and running the ioslevel command. It should indicate the ioslevel is now *$ ioslevel 2.1.0.0*.
- Restart previously running daemons and agents:
  1. Log on to the Virtual I/O Server as padmin user.

2. Complete the following command: `$ motd -overwrite` "<enter previous banner message>"
3. Start up any previously running daemons, such as FTP and Telnet.
4. Start up any previously running agents, such as ituam.

- Check for updates to the Virtual I/O Server. For instructions, see Virtual I/O Server support site.

  Remember: The Virtual I/O Server migration media is separate from the Virtual I/O Server installation media. Do not use the installation media for updates after you perform a migration. It does not contain updates and you will lose your current configuration. Only apply updates using the instructions from the Virtual I/O Server support site.

Related tasks

"Backing up the Virtual I/O Server to a remote file system by creating a mksysb image" on page 129
You can back up the Virtual I/O Server base code, applied fix packs, custom device drivers to support disk subsystems, and some user-defined metadata to a remote file system by creating a mksysb file.

Related information

Migrating the Virtual I/O Server from DVD when using the Integrated Virtualization Manager

Migrating the Virtual I/O Server from DVD when using the Integrated Virtualization Manager

Migrating the Virtual I/O Server from DVD

Find instructions for migrating the Virtual I/O Server from a DVD device that is attached to the Virtual I/O Server logical partition.

Before you start, ensure that the following statements are true:

- An HMC is attached to the managed system.
- A DVD optical device is assigned to the Virtual I/O Server logical partition.
- The Virtual I/O Server migration installation media is required.

  Note: The Virtual I/O Server migration installation media is separate from the Virtual I/O Server installation media.
- The Virtual I/O Server is currently at version 1.3, or later.
- The rootvg volume group has been assigned to the Virtual I/O Server
- Back up the mksysb image before migrating the Virtual I/O Server. Run the backupios command and save the mksysb image to a safe location.

Note: If you are using an Integrated Virtualization Manager (IVM) environment, see Migrating the Virtual I/O Server from DVD when using the Integrated Virtualization Manager.

To migrate the Virtual I/O Server from a DVD, follow these steps:

1. Activate the Virtual I/O Server logical partition using the HMC, version 7 (or later):
   a. Insert the **Virtual I/O Server migration DVD** into the DVD drive assigned to the Virtual I/O Server logical partition.
   b. In the HMC navigation area, expand **Systems Management** → **Servers**.
   c. Select the server on which the Virtual I/O Server logical partition is located.
   d. In the contents area, select the Virtual I/O Server logical partition.
   e. Click **Tasks** → **Operations** → **Activate**. The Activate Partition menu opens with a selection of logical partition profiles. Ensure that the correct profile is highlighted.
   f. Select **Open a terminal window or console session** to open a virtual terminal (vterm) window.
   g. Click **Advanced** to open the advanced options menu.
   h. For the boot mode, select **SMS**.
   i. Click **OK** to close the advanced options menu.

   j. Click **OK**. A virtual terminal window opens for the logical partition.
2. Select the boot device:
   a. Select **Select Boot Options** and press Enter.
   b. Select **Select Install/Boot Device** and press Enter.
   c. Select **CD/DVD** and press Enter.
   d. Select the device number that corresponds to the DVD and press Enter. You can also select **List all devices** and select the device number from a list and press Enter.
   e. Select **Normal mode boot**.
   f. Select **Yes** to exit SMS.
3. Install the Virtual I/O Server:
   a. Select the desired console and press Enter.
   b. Select a language for the BOS menus and press Enter.
   c. Select **Start Install Now with Default Settings** and press Enter. You can also verify the installation and system settings by typing `2` to select **Change/Show Installation Settings and Install**.

      Note: You should not have to change installation settings simply to select the migration installation method. If a previous version of the operating system exists, the installation method defaults to migration.
   d. Select **Continue with Install**. The system will reboot after the installation is complete.

After the migration is complete, the Virtual I/O Server logical partition is restarted to its preserved configuration prior to the migration installation. It is recommended that you perform the following tasks:

- Verify that migration was successful by checking results of the installp command and running the ioslevel command. It should indicate the ioslevel is now *$ ioslevel 2.1.0.0*.
- Restart previously running daemons and agents:
  1. Log on to the Virtual I/O Server as `padmin` user.
  2. Complete the following command: `$ motd -overwrite` ″<enter previous banner message>″
  3. Start any previously running daemons, such as FTP and Telnet.
  4. Start any previously running agents, such as ituam.
- Check for updates to the Virtual I/O Server. For instructions, see Virtual I/O Server support site.

  Remember: The Virtual I/O Server migration media is separate from the Virtual I/O Server installation media. Do not use the installation media for updates after you perform a migration. It does not contain updates and you will lose your current configuration. Only apply updates using the instructions from the Virtual I/O Server support site.

Related tasks

"Backing up the Virtual I/O Server to a remote file system by creating a mksysb image" on page 129
You can back up the Virtual I/O Server base code, applied fix packs, custom device drivers to support disk subsystems, and some user-defined metadata to a remote file system by creating a mksysb file.

Related information

Migrating the Virtual I/O Server from DVD when using the Integrated Virtualization Manager

Migrating the Virtual I/O Server from DVD when using the Integrated Virtualization Manager

Configuring the Virtual I/O Server

You need to configure virtual SCSI and virtual Ethernet devices on the Virtual I/O Server. Optionally, you can also configure virtual fibre channel adapters, Tivoli agents and clients, and configure the Virtual I/O Server as an LDAP client.

Important: Before you upgrade an existing POWER5 Virtual I/O Server (VIOS) to a POWER6 VIOS, you must configure the maximum virtual I/O slot number and any virtual Ethernet, virtual serial, or virtual SCSI adapters that use VIOS slots 0 through 10. Understand the applicable configuration rules, by reviewing VIOS configuration rules when upgrading an existing system.

Configuring virtual SCSI on the Virtual I/O Server

You can configure virtual SCSI devices by deploying a system plan, creating volume groups and logical volumes, and configuring the Virtual I/O Server to support SCSI-2 reserve functions.

Provisioning virtual disk resources occurs on the Virtual I/O Server. Physical disks owned by the Virtual I/O Server can either be exported and assigned to a client logical partition as a whole or can be partitioned into parts, such as logical volumes or files. These logical volumes and files can be exported as virtual disks to one or more client logical partitions. Therefore, by using virtual SCSI, you can share adapters as well as disk devices.

To make a physical volume, logical volume, or file available to a client logical partition requires that it be assigned to a virtual SCSI server adapter on the Virtual I/O Server. The SCSI client adapter is linked to a particular virtual SCSI server adapter in the Virtual I/O Server logical partition. The client logical partition accesses its assigned disks through the virtual SCSI client adapter. The Virtual I/O Server client adapter sees standard SCSI devices and LUNs through this virtual adapter. Assigning disk resources to a SCSI server adapter in the Virtual I/O Server effectively allocates resources to a SCSI client adapter in the client logical partition.

For information about SCSI devices that you can use, see the Virtual I/O Server Support for UNIX servers and Midrange servers Web site.

Creating the virtual target device on the Virtual I/O Server

Creating the virtual target device on the Virtual I/O Server maps the virtual SCSI adapter with the file, logical volume, tape, optical device or physical disk.

With the Virtual I/O Server version 2.1 and later, you can export the following types of physical devices:

- Virtual SCSI disk backed by a physical volume
- Virtual SCSI disk backed by a logical volume
- Virtual SCSI disk backed by a file
- Virtual SCSI optical backed by a physical optical device
- Virtual SCSI optical backed by a file
- Virtual SCSI tape backed by a physical tape device

After a virtual device is assigned to a client partition, the Virtual I/O Server must be available before the client logical partitions can access it.

Creating a virtual target device on a Virtual I/O Server that maps to a physical or logical volume, tape or physical optical device:

You can create a virtual target device on a Virtual I/O Server that maps the virtual SCSI adapter to a physical disk, tape, or physical optical device, or to a logical volume that is based on a volume group.

The following procedure can be repeated to provide additional virtual disk storage to any client logical partition.

Before you start, ensure the following statements are true:

1. At least one physical volume, tape, or optical device, or logical volume is defined on the Virtual I/O Server. For information, see “Logical volumes” on page 13.

2. The virtual adapters for the Virtual I/O Server and the client logical partitions are created. This usually occurs during the creation of the logical partition profile. For information about creating the logical partition, see Installing the Virtual I/O Server.
3. Be aware of the maximum transfer size limitation when you use AIX clients and physical devices. If you have an existing and active AIX client, and you want to add another virtual target device to the virtual SCSI server adapter used by that client, ensure that the max_transfer attribute is the same size or larger than the devices already in use.

Tip: If you are using the HMC, version 7 release 3.4.2 or later, you can use the HMC graphical interface to create a virtual target device on the Virtual I/O Server.

To create a virtual target device that maps a virtual SCSI server adapter to a physical device or logical volume, complete the following steps from the Virtual I/O Server command-line interface:

1. Use the lsdev command to ensure that the virtual SCSI adapter is available. For example, running `lsdev -virtual` returns results similar to the following:
   ```
   name     status     description
   ent3     Available  Virtual I/O Ethernet Adapter (l-lan)
   vhost0   Available  Virtual SCSI Server Adapter
   vhost1   Available  Virtual SCSI Server Adapter
   vsa0     Available  LPAR Virtual Serial Adapter
   vtscsi0  Available  Virtual Target Device - Logical Volume
   vtscsi1  Available  Virtual Target Device - File-backed Disk
   vtscsi2  Available  Virtual Target Device - File-backed Disk
   ```
2. To create a virtual target device, which maps the virtual SCSI server adapter to a physical device or logical volume, run the mkvdev command:
   ```
   mkvdev -vdev TargetDevice -vadapter VirtualSCSIServerAdapter
   ```

   Where:
   - *TargetDevice* is the name of the target device, as follows:
     - To map a logical volume to the virtual SCSI server adapter, use the name of the logical volume. For example, lv_4G.
     - To map a physical volume to the virtual SCSI server adapter, use `hdisk`*x*. For example, hdisk5.
     - To map an optical device to the virtual SCSI server adapter, use `cd`*x*. For example, cd0.
     - To map a tape device to a virtual SCSI adapter, use `rmt`*x*. For example, rmt1.
   - *VirtualSCSIServerAdapter* is the name of the virtual SCSI server adapter.

   Note: If needed, use the lsdev and lsmap -all commands to determine the target device and virtual SCSI server adapter that you want to map to one another.

   The storage is available to the client logical partition either the next time it starts, or the next time the appropriate virtual SCSI client adapter is probed (on a Linux logical partition), or configured (on an AIX logical partition), or appears as a either a DDXXX or DPHXXX device (on an IBM i partition).
3. View the newly created virtual target device by running the lsdev command. For example, running `lsdev -virtual` returns results similar to the following:
   ```
   name     status     description
   vhost3   Available  Virtual SCSI Server Adapter
   vsa0     Available  LPAR Virtual Serial Adapter
   vtscsi0  Available  Virtual Target Device - Logical Volume
   vttape0  Available  Virtual Target Device - Tape
   ```
4. View the logical connection between the newly created devices by running the lsmap command. For example, running `lsmap -vadapter vhost3` returns results similar to the following:
   ```
   SVSA      Physloc                    Client PartitionID
   ------------------------------------------------------
   vhost3    U9111.520.10DDEEC-V1-C20   0x00000000

   VTD                    vtscsi0
   ```

```
Status              Available
LUN                 0x8100000000000000
Backing device      lv_4G
Physloc
```

The physical location is a combination of the slot number, in this case 20, and the logical partition ID. The storage is now available to the client logical partition either the next time it starts, or the next time the appropriate virtual SCSI client adapter is probed, or configured.

If you later need to remove the virtual target device, you can do so by using the rmvdev command.

Related concepts

"Virtual SCSI sizing considerations" on page 60
Understand the processor and memory-sizing considerations when implementing virtual SCSI .

Related information

Creating a virtual disk for a VIOS logical partition using the HMC

Virtual I/O Server and Integrated Virtualization Manager commands

Creating a virtual target device on a Virtual I/O Server that maps to a file or logical volume:

You can create a virtual target device on a Virtual I/O Server that maps the virtual SCSI adapter to a file or a logical volume that is based on a storage pool.

The following procedure can be repeated to provide additional virtual disk storage to any client logical partition.

Before you start, ensure the following statements are true:

- The Virtual I/O Server is at version 1.5 or later. To update the Virtual I/O Server, see "Updating the Virtual I/O Server" on page 126.
- At least one file is defined in a file storage pool, or at least one logical volume is defined in a logical volume storage pool on the Virtual I/O Server. For information, see "Virtual storage" on page 18 and "Storage pools" on page 16.
- The virtual adapters for the Virtual I/O Server and the client logical partitions are created. This usually occurs during the creation of the logical partition profile. For information about creating the logical partition, see Installing the Virtual I/O Server.

Tip: If you are using the HMC, version 7 release 3.4.2 or later, you can use the HMC graphical interface to create a virtual target device on the Virtual I/O Server.

To create a virtual target device that maps a virtual SCSI server adapter to a file or logical volume, complete the following steps from the Virtual I/O Server command-line interface:

1. Use the lsdev command to ensure that the virtual SCSI adapter is available. For example, running `lsdev -virtual` returns results similar to the following:

   ```
   name     status     description
   ent3     Available  Virtual I/O Ethernet Adapter (l-lan)
   vhost0   Available  Virtual SCSI Server Adapter
   vhost1   Available  Virtual SCSI Server Adapter
   vsa0     Available  LPAR Virtual Serial Adapter
   vtscsi0  Available  Virtual Target Device - Logical Volume
   vtscsi1  Available  Virtual Target Device - File-backed Disk
   vtscsi2  Available  Virtual Target Device - File-backed Disk
   ```

2. To create a virtual target device, which maps the virtual SCSI server adapter to a file or logical volume, run the mkbdsp command:

   ```
   mkbdsp -sp StoragePool -bd BackingDevice -vadapter VirtualSCSIServerAdapter -tn TargetDeviceName
   ```

   Where:

   - *StoragePool* is the name of the storage pool that contains the file or logical volume to which you plan to map the virtual SCSI server adapter. For example, fbPool.
   - *BackingDevice* is the name of the file or logical volume to which you plan to map the virtual SCSI server adapter. For example, devFile.
   - *VirtualSCSIServerAdapter* is the name of the virtual SCSI server adapter. For example, vhost4.
   - *TargetDeviceName* is the name of the target device. For example, fbvtd1.

   The storage is available to the client logical partition either the next time it starts, or the next time the appropriate virtual SCSI client adapter is probed (on a Linux logical partition), or configured (on an AIX logical partition), or appears as a either a DDXXX or DPHXXX device (on an IBM i logical partition).

3. View the newly created virtual target device by running the lsdev command. For example, running `lsdev -virtual` returns results similar to the following:

   ```
   name     status     description
   vhost4   Available  Virtual SCSI Server Adapter
   vsa0     Available  LPAR Virtual Serial Adapter
   fbvtd1   Available  Virtual Target Device - File-backed Disk
   ```

4. View the logical connection between the newly created devices by running the lsmap command. For example, running `lsmap -vadapter vhost4` returns results similar to the following:

   ```
   SVSA     Physloc                     Client PartitionID
   -------------------------------------------------------
   vhost4   U9117.570.10C8BCE-V6-C2     0x00000000

   VTD              fbvtd1
   Status           Available
   LUN              0x8100000000000000
   Backing device   /var/vio/storagepools/fbPool/devFile
   Physloc
   ```

   The physical location is a combination of the slot number, in this case 2, and the logical partition ID. The virtual device can now be attached from the client logical partition.

If you later need to remove the virtual target device and backup device (file or logical volume), use the rmbdsp command. An option is available on the rmbdsp command to remove the virtual target device without removing the backup device. A backup device file is associated with a virtual target device by inode number rather than by file name, so do not change the inode number of a backing device file. The inode number might change if you alter a backup device file (using the AIX rm, mv, and cp commands), while the backup device file is associated with a virtual target device.

Related information

Creating a virtual disk for a VIOS logical partition using the HMC

Virtual I/O Server and Integrated Virtualization Manager commands

Creating a virtual target device on a Virtual I/O Server that maps to a file-backed virtual optical device:

You can create a virtual target device on a Virtual I/O Server that maps the virtual SCSI adapter to a file-backed virtual optical device.

The following procedure can be repeated to provide additional virtual disk storage to any client logical partition.

Before you start, complete the following steps:

1. Ensure that the Virtual I/O Server is at version 1.5 or later. To update the Virtual I/O Server, see “Updating the Virtual I/O Server” on page 126.

2. Ensure that the virtual adapters for the Virtual I/O Server and the client logical partitions are created. This usually occurs during the creation of the logical partition profile. For information about creating the logical partition, see "Installing the Virtual I/O Server and client logical partitions" on page 78.

Tip: If you are using the HMC, version 7 release 3.4.2 or later, you can use the HMC graphical interface to create a virtual target device on the Virtual I/O Server.

To create a virtual target device that maps a virtual SCSI server adapter to a file-backed virtual optical device, complete the following steps from the Virtual I/O Server command-line interface:

1. Use the lsdev command to ensure that the virtual SCSI adapter is available. For example, running `lsdev -virtual` returns results similar to the following:

   ```
   name     status     description
   ent3     Available  Virtual I/O Ethernet Adapter (l-lan)
   vhost0   Available  Virtual SCSI Server Adapter
   vhost1   Available  Virtual SCSI Server Adapter
   vsa0     Available  LPAR Virtual Serial Adapter
   vtscsi0  Available  Virtual Target Device - Logical Volume
   vtscsi1  Available  Virtual Target Device - File-backed Disk
   vtscsi2  Available  Virtual Target Device - File-backed Disk
   ```

2. To create a virtual target device, which maps the virtual SCSI server adapter to a file-backed virtual optical device, run the mkvdev command:

   ```
   mkvdev -fbo -vadapter VirtualSCSIServerAdapter
   ```

   where *VirtualSCSIServerAdapter* is the name of the virtual SCSI server adapter. For example, vhost1.

   Note: No backing device is specified when creating virtual target devices for file-backed virtual optical devices because the drive is considered to contain no media. For information about loading media into a file-backed optical drive, see the loadopt command.

   The optical device is available to the client logical partition either the next time it starts, or the next time the appropriate virtual SCSI client adapter is probed (on a Linux logical partition), or configured (on an AIX logical partition), or appears as an OPTXXX device (on an IBM i logical partition).

3. View the newly created virtual target device by running the lsdev command. For example, running `lsdev -virtual` returns results similar to the following:

   ```
   name     status     description
   vhost4   Available  Virtual SCSI Server Adapter
   vsa0     Available  LPAR Virtual Serial Adapter
   vtopt0   Available  Virtual Target Device - File-backed Optical
   ```

4. View the logical connection between the newly created devices by running the lsmap command. For example, running `lsmap -vadapter vhost1` returns results similar to the following:

   ```
   SVSA     Physloc                   Client PartitionID
   ---------------------------------------------------
   vhost1   U9117.570.10C8BCE-V6-C2  0x00000000

   VTD                vtopt0
   LUN                0x8200000000000000
   Backing device     Physloc
   ```

   The physical location is a combination of the slot number, in this case 2, and the logical partition ID. The virtual device can now be attached from the client logical partition.

You can use the loadopt command to load file-backed virtual optical media into the file-backed virtual optical device.

If you later need to remove the virtual target device, you can do so by using the rmvdev command.

Related information

Creating a virtual disk for a VIOS logical partition using the HMC

Virtual I/O Server and Integrated Virtualization Manager commands

Setting the reserve policy attributes of a device:

In some configurations, you must consider the reservation policy of the device on the Virtual I/O Server.

The following table explains the situations in which the reservation policy of the device on the Virtual I/O Server is important for systems that are managed by the Hardware Management Console (HMC) and the Integrated Virtualization Manager (IVM).

Table 31. Situations where the reservation policy of a device is important

HMC-managed systems	IVM-managed systems
• To use an Multipath I/O (MPIO) configuration at the client, none of the virtual SCSI devices on the Virtual I/O Server can be reserving the virtual SCSI device. • For Live Partition Mobility, the reserve attribute on the physical storage that is used by the mobile partition must be set to no reserve for the mobile partition to access it from both the source and destination servers. • For PowerVM Active Memory Sharing, the Virtual I/O Server automatically sets the reserve attribute on the physical volume to no reserve when you add a paging space device to the shared memory pool.	For Live Partition Mobility, the reserve attribute on the physical storage that is used by the mobile partition must be set to no reserve for the mobile partition to access it from both the source and destination servers.

In these configurations, you need to ensure that the **reserve_policy** attribute of the device is set to `no_reserve`.

1. From a Virtual I/O Server logical partition, list the disks (or paging space devices) to which the Virtual I/O Server has access. Run the following command:
   ```
   lsdev -type disk
   ```
2. To determine the reserve policy of a disk, run the following command, where *hdiskX* is the name of the disk that you identified in step 1. For example, hdisk5.
   ```
   lsdev -dev hdiskX -attr reserve_policy
   ```
   If the reserve_policy value is anything other than `no_reserve`, it must be changed so that you can use the disk in any of the above configurations. The results might look similar to the following:
   ```
   ..
   reserve_policy  no_reserve                        Reserve Policy                   True
   ```
3. To set the reserve_policy to `no_reserve`, run the following command, where *hdiskX* is the name of the disk for which you want to set the reserve_policy attribute to `no_reserve`.
   ```
   chdev -dev hdiskX -attr reserve_policy=no_reserve
   ```
4. Repeat this procedure from the other Virtual I/O Server logical partition. Although the reserve_policy attribute is an attribute of the device, each Virtual I/O Server saves the value of the attribute. You must set the reserve_policy attribute from both Virtual I/O Server logical partitions so that both Virtual I/O Server logical partitions recognize that the reserve_policy of the device is set to no_reserve.

Creating logical volume storage pools on a Virtual I/O Server

You can create a logical volume storage pool on a Virtual I/O Server using the Hardware Management Console or the mksp and mkbdsp commands.

Before you start, ensure that the Virtual I/O Server is at version 1.5 or later. To update the Virtual I/O Server, see "Updating the Virtual I/O Server" on page 126.

Tip: If you are using the HMC, version 7 release 3.4.2 or later, you can use the HMC graphical interface to create logical volume storage pools on the Virtual I/O Server.

Logical volume storage pools are volume groups, which are collections of one or more physical volumes. The physical volumes that comprise a logical volume storage pool can be of varying sizes and types.

To create a logical volume storage pool, complete the following steps from the Virtual I/O Server command-line interface:

1. Create a logical volume storage pool by running the mksp command:

   ```
   mksp -f dev_clients hdisk2 hdisk4
   ```

   In this example, the name of the storage pool is `dev_clients` and it contains `hdisk2` and `hdisk4`.
2. Define a logical volume, which will be visible as a disk to the client logical partition. The size of this logical volume will act as the size of disks that will be available to the client logical partition. Use the mkbdsp command to create a 11 GB logical volume called `dev_dbsrv` as follows:

   ```
   mkbdsp -sp dev_clients 11G -bd dev_dbsrv
   ```

   If you also want to create a virtual target device, which maps the virtual SCSI server adapter to the logical volume, add `-vadapter` `vhost`*`x`* to the end of the command. For example:

   ```
   mkbdsp -sp dev_clients 11G -bd dev_dbsrv -vadapter vhost4
   ```

Related information

Creating storage pools on a Virtual I/O Server by using the HMC

Virtual I/O Server and Integrated Virtualization Manager commands

Creating file storage pools on a Virtual I/O Server

You can create a file storage pool on a Virtual I/O Server using the mksp and mkbdsp commands.

Before you start, ensure that the Virtual I/O Server is at version 1.5 or later. To update the Virtual I/O Server, see "Updating the Virtual I/O Server" on page 126.

Tip: If you are using the HMC, version 7 release 3.4.2 or later, you can use the HMC graphical interface to create file storage pools on the Virtual I/O Server.

File storage pools are created within a parent logical volume storage pool and contain a logical volume containing a file system with files.

To create a file storage pool, complete the following steps from the Virtual I/O Server command-line interface:

1. Create a file storage pool by running the mksp command:

   ```
   mksp -fb dev_fbclt -sp dev_clients -size 7g
   ```

   In this example, the name of the file storage pool is `dev_fbclt` and the parent storage pool is `dev_clients`.
2. Define a file, which will be visible as a disk to the client logical partition. The size of the file determines the size of the disk presented to the client logical partition. Use the mkbdsp command to create a 3 GB file called `dev_dbsrv` as follows:

   ```
   mkbdsp -sp dev_fbclt 3G -bd dev_dbsrv
   ```

   If you also want to create a virtual target device, which maps the virtual SCSI server adapter to the file, add `-vadapter` `vhost`*`x`* to the end of the command. For example:

```
mkbdsp -sp dev_fbclt 3G -bd dev_dbsrv -vadapter vhost4
```

Related information

Creating storage pools on a Virtual I/O Server by using the HMC

Virtual I/O Server and Integrated Virtualization Manager commands

Creating the virtual media repository on a Virtual I/O Server

You can create the virtual media repository on a Virtual I/O Server using the mkrep command.

Before you start, ensure that the Virtual I/O Server is at version 1.5 or later. To update the Virtual I/O Server, see "Updating the Virtual I/O Server" on page 126.

The virtual media repository provides a single container to store and manage file-backed virtual optical media files. Media stored in the repository can be loaded into file-backed virtual optical devices for exporting to client partitions.

Only one repository can be created within a Virtual I/O Server.

Tip: If you are using the HMC, version 7 release 3.4.2 or later, you can use the HMC graphical interface to create a virtual media repository on the Virtual I/O Server.

To create the virtual media repository from the Virtual I/O Server command-line interface, run the mkrep command:

```
mkrep -sp prod_store -size 6g
```

In this example, the name of the parent storage pool is `prod_store`.

Related information

Changing optical devices by using the Hardware Management Console

Virtual I/O Server and Integrated Virtualization Manager commands

Creating volume groups and logical volumes on a Virtual I/O Server

You can create logical volumes and volume groups on a Virtual I/O Server using the mkvg and mklv commands.

If you are using the HMC, version 7 release 3.4.2 or later, you can use the HMC graphical interface to create volume groups and logical volumes on a Virtual I/O Server.

Otherwise, use the mklv command from the Virtual I/O Server command-line interface. To create the logical volume on a separate disk, you must first create a volume group and assign one or more disks by using the mkvg command.

1. Create a volume group and assign a disk to this volume group by using the mkvg command. In this example, the name of the volume group is `rootvg_clients`

   ```
   mkvg -f -vg rootvg_clients hdisk2
   ```

2. Define a logical volume, which will be visible as a disk to the client logical partition. The size of this logical volume will act as the size of disks that will be available to the client logical partition. Use the mklv command to create a 2 GB logical volume as follows:

   ```
   mklv -lv rootvg_dbsrv rootvg_clients 2G
   ```

Related information

Changing a physical volume for a VIOS logical partition using the HMC

Changing a storage pool for a VIOS logical partition using the HMC

Configure the Virtual I/O Server to support SCSI-2 reserve functions

Understand the virtual SCSI setup requirements to support applications using SCSI reserve and release.

Virtual I/O Server versions 1.3 and later provide support for applications that are enabled to use SCSI-2 reserve functions that are controlled by the client logical partition. Typically, SCSI reserve and release is used in clustered environments where contention for SCSI disk resources might require greater control. To ensure that Virtual I/O Server supports these environments, configure the Virtual I/O Server to support SCSI-2 reserve and release. If the applications you are using provide information about the policy to use for the SCSI-2 reserve functions on the client logical partition, follow those procedures for setting the reserve policy.

Complete the following tasks to configure the Virtual I/O Server to support SCSI-2 reserve environments:

1. Configure the Virtual I/O Server reserve_policy for single_path, using the following command:

   ```
   chdev -dev1 hdiskN -attr reserve_policy=single_path
   ```

   Note: Perform this task when the device is not in use. If you run this command while the device is open or in use, then you must use the **-perm** flag with this command. If you use the **-perm** flag, the changes do not take effect until the device is unconfigured and reconfigured.
2. Configure the client_reserve feature on the Virtual I/O Server.
   - If you are creating a virtual target device, use the following command:

     ```
     mkvdev -vdev hdiskN -vadapter vhostN -attr client_reserve=yes
     ```

     where *hdiskN* is the virtual target device name and *vhostN* is the virtual SCSI server adapter name.
   - If the virtual target device has already been created, use the following command:

     ```
     chdev -dev vtscsiN -attr client_reserve=yes
     ```

     where *vtscsiN* is the virtual device name.
3. On the Virtual client, complete the following steps to configure the SCSI reserve and release support for the virtual disk backed by the physical disk that you configured in step 1:
   a. Set the reserve policy on the Virtual client to single_path, using the following command:

      ```
      chdev -a reserve_policy=single_path -1 hdiskN
      ```

      where *hdiskN* is the virtual disk name

      Note: Perform this task when the device is not in use. If you run this command while the device is open or in use, then you must use the **-p** flag. In that case, the changes do not take effect until the device is unconfigured and reconfigured.

   b. Set the hcheck_cmd attribute so that the MPIO code uses the inquiry option. If the hcheck_cmd attribute is set to **test unit ready** and the backing device is reserved, then *test unit ready* will fail and log an error on the client.

      ```
      chdev -a hcheck_cmd=inquiry -1 hdiskN
      ```

      where *hdiskN* is the virtual disk name.

Identifying exportable disks

To export a physical volume as a virtual device, the physical volume must have an IEEE volume attribute, a unique identifier (UDID), or a physical identifier (PVID).

To identify exportable disks, complete the following steps:

1. Determine whether a device has an IEEE volume attribute identifier by running the following command from the Virtual I/O Server command line:

   ```
   lsdev -dev hdiskX -attr
   ```

   Disks with an IEEE volume attribute identifier have a value in the `ieee_volname` field. Output similar to the following is displayed:

   ```
   ...
   cache_method   fast_write                       Write Caching method
      False
   ieee_volname   600A0B800012DD0D00000AB441ED6AC IEEE Unique volume name
      False
   lun_id         0x001a000000000000               Logical Unit Number
      False
   ...
   ```

   If the `ieee_volname` field does not appear, then the device does not have an IEEE volume attribute identifier.
2. If the device does not have an IEEE volume attribute identifier, then determine whether the device has a UDID by completing the following steps:
   a. Type `oem_setup_env`.
   b. Type `odmget -qattribute=`*`unique_id`* `CuAt`. The disks that have a UDID are listed. Output similar to the following is displayed:

   ```
   CuAt:
    name = "hdisk1"
    attribute = "unique_id"
    value = "2708ECVBZ1SC10IC35L146UCDY10-003IBMscsi"
    type = "R"
    generic = ""
    rep = "nl"
    nls_index = 79

   CuAt:
    name = "hdisk2"
    attribute = "unique_id"
    value = "210800038FB50AST373453LC03IBMscsi"
    type = "R"
    generic = ""
    rep = "nl"
    nls_index = 79
   ```

   Devices in the list that are accessible from other Virtual I/O Server partitions can be used in virtual SCSI MPIO configurations.
   c. Type `exit`.
3. If the device does not have either an IEEE volume attribute identifier or a UDID, then determine whether the device has a PVID by running the following command:

   ```
   lspv
   ```

   The disks and their respective PVIDs are listed. Output similar to the following is displayed:

   ```
   NAME        PVID                  VG       STATUS
   hdisk0      00c5e10c1608fd80      rootvg   active
   hdisk1      00c5e10cf7eb2195      rootvg   active
   hdisk2      00c5e10c44df5673      None
   hdisk3      00c5e10cf3ba6a9a      None
   hdisk4      none                  None
   ```
4. If the device does not have either an IEEE volume attribute identifier, a UDID, or a PVID, then complete one of the following tasks to assign an identifier:

a. Upgrade your vendor software and then repeat this entire procedure, Identifying exportable disks, from the beginning. The latest versions of some vendor software include support for identifying devices using a UDID. Before upgrading, ensure that you preserve any virtual SCSI devices that you created when using the versions of the software that did not support identifying devices using a UDID. For information and upgrade instructions, see the documentation provided by your vendor software.

b. If the upgraded vendor software does not produce a UDID or IEEE volume attribute identifier, then put a PVID on the physical volume by running the following command:

```
chdev -dev hdiskX -attr pv=yes
```

Configuring virtual Ethernet on the Virtual I/O Server

You can configure virtual Ethernet devices by deploying a system plan, create and configure a Shared Ethernet Adapter, and configure a Link Aggregation device.

Creating a virtual Ethernet adapter using HMC version 7

You can create a virtual Ethernet adapter on a Virtual I/O Server so that client logical partitions can access the external network without having to own a physical Ethernet adapter.

If you plan to use a Shared Ethernet Adapter with a Host Ethernet Adapter (or Integrated Virtual Ethernet), ensure that the Logical Host Ethernet Adapter (LHEA) on the Virtual I/O Server is set to promiscuous mode. For instructions, see “Setting the LHEA to promiscuous mode” on page 105.

To create a virtual Ethernet adapter on the Virtual I/O Server using the Hardware Management Console (HMC), version 7 or later, complete the following steps:

1. In the navigation area, expand **Systems Management** → **Servers** and select the server on which the Virtual I/O Server logical partition is located.
2. In the contents are, select the Virtual I/O Server logical partition.
3. Click **Tasks** and select **Configuration** → **Manage Profiles**. The Managed Profiles page is displayed.
4. Select the profile in which you want to create the Shared Ethernet Adapter and click **Actions** → **Edit**. The Logical Partition Profile Properties page is displayed.
5. Click the **Virtual Adapters** tab.
6. Click **Actions** → **Create** → **Ethernet adapter**.
7. Select **IEEE 802.1Q-compatible adapter**.
8. If you are using multiple VLANs, add any additional VLAN IDs for the client logical partitions that must communicate with the external network using this virtual adapter.
9. Select **Access external network** to use this adapter as a gateway between VLANs and an external network. This Ethernet adapter is configured as part of the Shared Ethernet Adapter.
10. If you are not using Shared Ethernet Adapter failover, you can use the default trunk priority. If you are using Shared Ethernet Adapter failover, then set the trunk priority for the primary share Ethernet adapter to a lower number than that of the backup Shared Ethernet Adapter.
11. When you are finished, click **OK**.
12. Assign or create one of the following real adapters:
    - Assign a physical Ethernet adapter to the Virtual I/O Server.
    - If you plan to aggregate more than one physical Ethernet adapter into a Link Aggregation or EtherChannel device, then assign multiple physical Ethernet adapters to the Virtual I/O Server.
    - If you plan to use the Shared Ethernet Adapter with a Host Ethernet Adapter, then create an LHEA for the Virtual I/O Server logical partition.
13. Click **OK** to exit the Logical Partition Profile Properties page.
14. Click **Close** to exit the Managed Profiles page.
15. Repeat this procedure for additional Shared Ethernet Adapters that you require.

When you are finished, configure the Shared Ethernet Adapter using the Virtual I/O Server command-line interface or the Hardware Management Console graphical interface, version 7 release 3.4.2 or later.

Related tasks

"Configuring a Shared Ethernet Adapter"
Find instructions for configuring Shared Ethernet Adapters.

Setting the LHEA to promiscuous mode:

To use a Shared Ethernet Adapter with a Host Ethernet Adapter (or Integrated Virtual Ethernet), you must set the Logical Host Ethernet Adapter (LHEA) to promiscuous mode.

Before you start, use the Hardware Management Console (HMC) to determine the physical port of the Host Ethernet Adapter that is associated with the Logical Host Ethernet port. Determine this information for the Logical Host Ethernet port that is the real adapter of the Shared Ethernet Adapter on the Virtual I/O Server. You can find this information in the partition properties of the Virtual I/O Server, and the managed system properties of the server on which the Virtual I/O Server is located.

To set the Logical Host Ethernet port (that is the real adapter of the Shared Ethernet Adapter) to promiscuous mode, complete the following steps using the HMC:

1. In the navigation area, expand **Systems Management** and click **Servers**.
2. In the contents area, select the server on which the Virtual I/O Server logical partition is located.
3. Click **Tasks** and select **Hardware (information)** → **Adapters** → **Host Ethernet**. The HEAs page is shown.
4. Select the physical location code of the Host Ethernet Adapter.
5. Select the physical port associated with the Logical Host Ethernet port on the Virtual I/O Server logical partition, and click **Configure**. The HEA Physical Port Configuration page is shown.
6. Select **VIOS** in the Promiscuous LPAR field.
7. Click **OK** twice to return to the contents area.

Configuring a Shared Ethernet Adapter

Find instructions for configuring Shared Ethernet Adapters.

Before you can configure a Shared Ethernet Adapter, you must first create the adapter using the Hardware Management Console (HMC). For instructions, see "Creating a virtual Ethernet adapter using HMC version 7" on page 104.

To configure a Shared Ethernet Adapter using the HMC, version 7 release 3.4.2 or later, see Creating a shared Ethernet adapter for a Virtual I/O Server logical partition using the Hardware Management Console.

To configure a Shared Ethernet Adapter using versions prior to the HMC, version 7 release 3.4.2, complete the following steps from the Virtual I/O Server command-line interface:

1. Verify that the virtual Ethernet trunk adapter is available by running the following command:

   ```
   lsdev -virtual
   ```

2. Identify the appropriate physical Ethernet adapter that will be used to create the Shared Ethernet Adapter by running the following command:

   ```
   lsdev -type adapter
   ```

   Notes:

   - Ensure that TCP/IP is not configured on the interface for the physical Ethernet adapter. If TCP/IP is configured, the mkvdev command in the next step fails.
   - You can also use a Link Aggregation, or EtherChannel, device as the Shared Ethernet Adapter.

- If you plan to use the Host Ethernet Adapter or Integrated Virtual Ethernet with the Shared Ethernet Adapter, ensure that you use the Logical Host Ethernet Adapter to create the Shared Ethernet Adapter.

3. Configure the Shared Ethernet Adapter by running the following command:

   ```
   mkvdev -sea target_device -vadapter virtual_ethernet_adapters \
   -default DefaultVirtualEthernetAdapter -defaultid SEADefaultPVID
   ```

   Where:

   target_device
   The physical adapter being used as part of the Shared Ethernet Adapter device.

   virtual_ethernet_adapters
   The virtual Ethernet adapter or adapters that will use the Shared Ethernet Adapter.

   DefaultVirtualEthernetAdapter
   The default virtual Ethernet adapter used to handle untagged packets. If you have only one virtual Ethernet adapter for this logical partition, use it as the default.

   SEADefaultPVID
   The PVID associated with your default virtual Ethernet adapter.

   For example, to create Shared Ethernet Adapter `ent3` with `ent0` as the physical Ethernet adapter (or Link Aggregation) and `ent2` as the only virtual Ethernet adapter (defined with a PVID of 1), type the following command:

   ```
   mkvdev -sea ent0 -vadapter ent2 -default ent2 -defaultid 1
   ```

4. Verify that the Shared Ethernet Adapter was created by running the following command:

   ```
   lsdev -virtual
   ```

5. Do you plan to access the Virtual I/O Server from the network with the physical device used to create the Shared Ethernet Adapter?
   - Yes: Go to step 6.
   - No: You are finished with this procedure and do not need to complete the remaining steps.

6. Do you plan to set bandwidth apportioning by defining a Quality of Service (QoS)?
   - Yes: Go to step 11 to enable the Shared Ethernet Adapter device to prioritize traffic.
   - No: Go to step 9 to configure a TCP/IP connection.

7. Do you plan to define IP addresses on any VLANs other than the VLAN specified by the PVID of the Shared Ethernet Adapter?
   - Yes: Go to step 8 to create VLAN pseudo-devices.
   - No: Go to step 9 on page 107 to configure a TCP/IP connection.

8. To configure VLAN pseudo-devices, complete the following steps:

   a. Create a VLAN pseudo-device on the Shared Ethernet Adapter by running the following command:

      ```
      mkvdev -vlan TargetAdapter -tagid TagID
      ```

      Where:
      - *TargetAdapter* is the Shared Ethernet Adapter.
      - *TagID* is the VLAN ID that you defined when creating the virtual Ethernet adapter associated with the Shared Ethernet Adapter.

      For example, to create a VLAN pseudo-device using the Shared Ethernet Adapter `ent3` that you just created with a VLAN ID of 1, type the following command:

      ```
      mkvdev -vlan ent3 -tagid 1
      ```

   b. Verify that the VLAN pseudo-device was created by running the following command:

      ```
      lsdev -virtual
      ```

c. Repeat this step for any additional VLAN pseudo-devices that you need.

9. Run the following command to configure the first TCP/IP connection. The first connection must be on the same VLAN and logical subnet as the default gateway.

```
mktcpip -hostname Hostname -inetaddr Address -interface Interface -netmask \
SubnetMask -gateway Gateway -nsrvaddr NameServerAddress -nsrvdomain Domain
```

   Where:

   - *Hostname* is the host name of the Virtual I/O Server
   - *Address* is the IP address you want to use for the TCP/IP connection
   - *Interface* is the interface associated with either the Shared Ethernet Adapter device or a VLAN pseudo-device. For example, if the Shared Ethernet Adapter device is ent3, the associated interface is en3.
   - *Subnetmask* is the subnet mask address for your subnet.
   - *Gateway* is the gateway address for your subnet.
   - *NameServerAddress* is the address of your domain name server.
   - *Domain* is the name of your domain.

   If you do not have additional VLANs, then you are finished with this procedure and do not need to complete the remaining step.

10. Run the following command to configure additional TCP/IP connections:

```
chdev -dev interface -perm -attr netaddr=IPaddress -attr netmask=netmask
-attr state=up
```

   When using this command, enter the interface (en*X*) associated with either the Shared Ethernet Adapter device or VLAN pseudo-device.

11. Enable the Shared Ethernet Adapter device to prioritize traffic. Client logical partitions must insert a VLAN priority value in their VLAN header. For AIX clients, a VLAN pseudo-device must be created over the Virtual I/O Ethernet Adapter, and the VLAN priority attribute must be set (the default value is 0). Do the following steps to enable traffic prioritization on an AIX client:

    a. Set the Shared Ethernet Adapter qos_mode attribute to either strict or loose mode. Use one of the following commands: `chdev -dev <SEA device name> -attr qos_mode=strict` or `chdev -dev <SEA device name> -attr qos_mode=loose`. For more information about the modes, see Shared Ethernet Adapter.

    b. From the HMC, create a Virtual I/O Ethernet Adapter for the AIX client with all of the tagged VLANs that are required (specified in the `Additional VLAN ID` list). Packets sent over the default VLAN ID (specified in the **Adapter ID** or **Virtual LAN ID** field) will not be tagged as VLAN; therefore, a VLAN priority value cannot be assigned to them.

    c. On the AIX client, run the **smitty vlan** command.

    d. Select **Add a VLAN**.

    e. Select the name of the Virtual I/O Ethernet Adapter created in step 1.

    f. In the VLAN Tag ID attribute, specify one of the tagged VLANs that are configured on the Virtual I/O Ethernet Adapter that you created in step 1.

    g. Specify an attribute value (0 - 7) in the VLAN Priority attribute, which corresponds to the importance the VIOS should give to the traffic sent over that VLAN pseudo-device.

    h. Configure the interface over the VLAN pseudo-device created in step 6.

Traffic sent over the interface created in step 7 will be tagged as VLAN and its VLAN header will have the VLAN priority value specified in step 6. When this traffic is bridged by a Shared Ethernet Adapter that has been enabled for bandwidth apportioning, the VLAN priority value is used to determine how quickly it should be sent in relation to other packets at different priorities.

The Shared Ethernet Adapter is now configured. After you configure the TCP/IP connections for the virtual adapters on the client logical partitions using the client logical partitions' operating systems, those logical partitions can communicate with the external network.

Related concepts

"Shared Ethernet Adapter failover" on page 72
Shared Ethernet Adapter failover provides redundancy by configuring a backup Shared Ethernet Adapter on a different Virtual I/O Server logical partition that can be used if the primary Shared Ethernet Adapter fails. The network connectivity in the client logical partitions continues without disruption.

"Shared Ethernet Adapters" on page 25
With Shared Ethernet Adapters on the Virtual I/O Server logical partition, virtual Ethernet adapters on client logical partitions can send and receive outside network traffic.

Related information

➪ Creating a shared Ethernet adapter for a VIOS logical partition using the HMC

➪ Virtual I/O Server and Integrated Virtualization Manager commands

Configuring a Link Aggregation or EtherChannel device

Configure a Link Aggregation device, also called an EtherChannel device, by using the mkvdev command. A Link Aggregation device can be used as the physical Ethernet adapter in the Shared Ethernet Adapter configuration.

Configure a Link Aggregation device by typing the following command:

```
mkvdev -lnagg TargetAdapter ... [-attr Attribute=Value ...]
```

For example, to create Link Aggregation device `ent5` with physical Ethernet adapters `ent3`, `ent4`, and backup adapter `ent2`, type the following:

```
mkvdev -lnagg ent3,ent4 -attr backup_adapter=ent2
```

After the Link Aggregation device is configured, you can add adapters to it, remove adapters from it, or modify its attributes using the cfglnagg command.

Assigning the virtual fibre channel adapter to a physical fibre channel adapter

To enable N-Port ID Virtualization (NPIV) on managed systems, connect the virtual fibre channel adapter on the Virtual I/O Server logical partition to a physical port on a physical fibre channel adapter.

Before you start, verify that the following statements are true:

- Verify that you have created the virtual fibre channel adapters on the Virtual I/O Server logical partition and associated them with virtual fibre channel adapters on the client logical partition.
- Verify that you have created the virtual fibre channel adapters on each client logical partition and associated them with a virtual fibre channel adapter on the Virtual I/O Server logical partition.

After the virtual fibre channel adapters are created, you need to connect the virtual fibre channel adapter on the Virtual I/O Server logical partition to the physical ports of the physical fibre channel adapter. The physical fibre channel adapter should be connected to the physical storage that you want the associated client logical partition to access.

Tip: If you are using the HMC, version 7 release 3.4.2 or later, you can use the HMC graphical interface to assign the virtual fibre channel adapter on a Virtual I/O Server to a physical fibre channel adapter.

To assign the virtual fibre channel adapter to a physical port on a physical fibre channel adapter, complete the following steps from the Virtual I/O Server command-line interface:

1. Use the lsnports command to display information for the available number of NPIV ports and available worldwide port names (WWPNs). For example, running lsnports returns results similar to the following:

   ```
   Name      Physloc                     fabric    tports    aports    swwpns    awwpns
   -----------------------------------------------------------------------------------
   fcs0     U789D.001.DQDMLWV-P1-C1-T1  1         64        64         2048     2047
   fcs1     U787A.001.DPM0WVZ-P1-C1-T2  1         63        62         504      496
   ```

   Note: If there are no NPIV ports in the Virtual I/O Server logical partition, the error code `E_NO_NPIV_PORTS(62)` is displayed.
2. To connect the virtual fibre channel adapter on the Virtual I/O Server logical partition to a physical port on a physical fibre channel adapter, run the vfcmap command: `vfcmap -vadapter` ***`virtual fibre channel adapter`*** `-fcp` ***`fibre channel port name`*** where:
   - *Virtual fibre channel adapter* is the name of the virtual fibre channel adapter created on the Virtual I/O Server logical partition.
   - *Fibre channel port name* is the name of the physical fibre channel port.

     Note: If no parameter is specified with the -fcp flag, the command unmaps the virtual fibre channel adapter from the physical fibre channel port.
3. Use the lsmap command to display the mapping between virtual host adapters and the physical devices to which they are backed. To list NPIV mapping information, type: `lsmap -all -npiv`. The system displays a message similar to the following:

   ```
   Name      Physloc                     ClntID    ClntName  ClntOS
   -------------------------------------------------------------
   vfchost0  U8203.E4A.HV40026-V1-C12   1         HV-40026   AIX

   Status:NOT_LOGGED_IN
   FC name:fcs0      FC loc code:U789C.001.0607088-P1-C5-T1
   Ports logged in:0
   Flags:1 <not_mapped, not_connected>
   VFC client name:    VFC client DRC:
   ```

When you are finished, consider the following tasks:

- For each logical partition, verify that both WWPNs are assigned to the same physical storage and have the same level of access on the storage area network (SAN). For instructions, see the IBM System Storage SAN Volume Controller.

  Note: To determine the WWPNs that are assigned to a logical partition, use the Hardware Management Console (HMC) to view the partition properties or partition profile properties of the client logical partition.
- If you later need to remove the connection between the virtual fibre channel adapter created on the Virtual I/O Server logical partition and the physical port, you can do so by using the vfcmap command and not specifying a parameter for the **-fcp** flag.

Related information

Configuring a virtual fibre channel adapter

Changing virtual fibre channel by using the Hardware Management Console

Virtual I/O Server and Integrated Virtualization Manager commands

Configuring the IBM Tivoli agents and clients on the Virtual I/O Server

You can configure and start the IBM Tivoli Monitoring agent, IBM Tivoli Usage and Accounting Manager, the IBM Tivoli Storage Manager client, and the IBM Tivoli TotalStorage Productivity Center agents.

Related concepts

"IBM Tivoli software and the Virtual I/O Server" on page 36
Learn about integrating the Virtual I/O Server into your Tivoli environment for IBM Tivoli Application Dependency Discovery Manager, IBM Tivoli Monitoring, IBM Tivoli Storage Manager, IBM Tivoli Usage and Accounting Manager, IBM Tivoli Identity Manager, and IBM TotalStorage Productivity Center.

Related information

cfgsvc command

Configuring the IBM Tivoli Monitoring agent

You can configure and start the IBM Tivoli Monitoring agent on the Virtual I/O Server.

With Tivoli Monitoring System Edition for System p, you can monitor the health and availability of multiple IBM System p servers (including the Virtual I/O Server) from the Tivoli Enterprise Portal. IBM Tivoli Monitoring System Edition for System p gathers data from the Virtual I/O Server, including data about physical volumes, logical volumes, storage pools, storage mappings, network mappings, real memory, processor resources, mounted file system sizes, and so on. From the Tivoli Enterprise Portal, you can view a graphical representation of the data, use predefined thresholds to alert you on key metrics, and resolve issues based on recommendations provided by the Expert Advice feature of Tivoli Monitoring.

Before you start, complete the following tasks:

- Ensure that the Virtual I/O Server is running fix pack 8.1.0. For instructions, see "Updating the Virtual I/O Server" on page 126.
- Verify that you are a super administrator of the HMC.
- Verify that you are the prime administrator of the Virtual I/O Server.

To configure and start the monitoring agent, complete the following steps:

1. List all of the available monitoring agents using the **lssvc** command. For example,

   ```
   $lssvc
   ITM_base
   ```

2. Based on the output of the **lssvc** command, decide which monitoring agent you want to configure. For example, `ITM_base`
3. List all of the attributes that are associated with the monitoring agent using the **cfgsvc** command. For example:

   ```
   $cfgsvc –ls ITM_base
    HOSTNAME
   RESTART_ON_REBOOT
   MANAGING_SYSTEM
   ```

4. Configure the monitoring agent with its associated attributes using the **cfgsvc** command:

   ```
   cfgsvc ITM_agent_name -attr Restart_On_Reboot=value hostname=name_or_address1
   managing_system=name_or_address2
   ```

   Where:

   - *ITM_agent_name* is the name of the monitoring agent. For example, ITM_base.
   - *value* must be either `TRUE` of `FALSE` as follows:
     - `TRUE`: *ITM_agent_name* restarts whenever the Virtual I/O Server restarts
     - `FALSE`: *ITM_agent_name* does not restart whenever the Virtual I/O Server restarts
   - *name_or_address1* is either the hostname or IP address of the Tivoli Enterprise Monitoring Server (TEMS) server to which *ITM_agent_name* sends data.
   - *name_or_address2* is either the hostname of IP address of the Hardware Management Console (HMC) attached to the managed system on which the Virtual I/O Server with the monitoring agent is located.

For example:

```
cfgsvc ITM_base –attr Restart_On_Reboot=TRUE hostname=tems_server managing_system=hmc_console
```

In this example, the ITM_base monitoring agent is configured to send data to tems_server, and to restart whenever the Virtual I/O Server restarts.

5. Start the monitoring agent using the **startsvc** command. For example:

   ```
   startsvc ITM_base
   ```

6. From the HMC, complete the following steps so that the monitoring agent can gather information from the HMC.

   Note: After you configure a secure shell connection for one monitoring agent, you do not need to configure it again for any additional agents.

   a. Determine the name of the managed system on which the Virtual I/O Server with the monitoring agent is located.

   b. Obtain the public key for the Virtual I/O Server by running the following command:

      ```
      viosvrcmd -m managed_system_name -p vios_name -c "cfgsvc -key ITM_agent_name"
      ```

      Where:

      - *managed_system_name* is the name of the managed system on which the Virtual I/O Server with the monitoring agent or client is located.
      - *vios_name* is the name of the Virtual I/O Server logical partition (with the monitoring agent) as defined on the HMC.
      - *ITM_agent_name* is the name of the monitoring agent. For example, `ITM_base`.

   c. Update the authorized_key2 file on the HMC by running the mkauthkeys command:

      ```
      mkauthkeys --add public_key
      ```

      where *public_key* is the output from the viosvrcmd command in step 6b.

   For example:

   ```
   $ viosvrcmd -m commo126041 -p VIOS7 -c "cfgsvc ITM_base -key"
   ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAvjDZ
     sS0guWzfzfp9BbweG0QMXv1tbDrtyWsgPbA2ExHA+xduWA51K0oFGarK2F
     C7e7NjKW+UmgQbrh/KSyKKwozjp4xWGNGhLmfan85ZpFR7wy9UQG1bLgXZ
     xYrY7yyQQQODjvwosWAfzkjpG3iW/xmWD5PKLBmob2QkKJbxjne+wqGwHT
     RYDGIiyhCBIdfFaLZgkXTZ2diZ98rL8LIv3qb+TsM1B28AL4t+1OGGeW24
     2lsB+8p4kamPJCYfKePHo67yP4NyKyPBFHY3TpTrca4/y1KEBT0Va3Pebr
     5JEIUvWYs6/RW+bUQk1Sb6eYbcRJFHhN5l3F+ofd0vj39zwQ== root@vi
     os7.vios.austin.ibm.com
   $ mkauthkeys --add 'ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAvjDZ
     sS0guWzfzfp9BbweG0QMXv1tbDrtyWsgPbA2ExHA+xduWA51K0oFGarK2F
     C7e7NjKW+UmgQbrh/KSyKKwozjp4xWGNGhLmfan85ZpFR7wy9UQG1bLgXZ
     xYrY7yyQQQODjvwosWAfzkjpG3iW/xmWD5PKLBmob2QkKJbxjne+wqGwHT
     RYDGIiyhCBIdfFaLZgkXTZ2diZ98rL8LIv3qb+TsM1B28AL4t+1OGGeW24
     2lsB+8p4kamPJCYfKePHo67yP4NyKyPBFHY3TpTrca4/y1KEBT0Va3Pebr
     5JEIUvWYs6/RW+bUQk1Sb6eYbcRJFHhN5l3F+ofd0vj39zwQ== root@vi
     os7.vios.austin.ibm.com'
   ```

When you are finished, you can view the data gathered by the monitoring agent from the Tivoli Enterprise Portal.

Related information

Tivoli Monitoring 6.1 documentation

Tivoli Monitoring Virtual I/O Server Premium Agent User's Guide

Configuring the IBM Tivoli Usage and Accounting Manager agent

You can configure and start the IBM Tivoli Usage and Accounting Manager agent on the Virtual I/O Server.

With Virtual I/O Server 1.4, you can configure the IBM Tivoli Usage and Accounting Manager agent on the Virtual I/O Server. Tivoli Usage and Accounting Manager helps you track, allocate, and invoice your IT costs by collecting, analyzing, and reporting on the actual resources used by entities such as cost centers, departments, and users. Tivoli Usage and Accounting Manager can gather data from multi-tiered datacenters that include Windows, AIX, Virtual I/O Server, HP/UX Sun Solaris, Linux, IBM i, and VMware.

Before you start, ensure that the Virtual I/O Server is installed. The Tivoli Usage and Accounting Manager agent is packaged with the Virtual I/O Server and is installed when the Virtual I/O Server is installed. For instructions, see "Installing the Virtual I/O Server and client logical partitions" on page 78.

To configure and start the Tivoli Usage and Accounting Manager agent, complete the following steps:

1. Optional: Add optional variables to the A_config.par file to enhance data collection. The A_config.par file is located at /home/padmin/tivoli/ituam/A_config.par. For more information about additional data collectors available for the ITUAM agent on the Virtual I/O Server, see the IBM Tivoli Usage and Accounting Manager Information Center.
2. List all of the available Tivoli Usage and Accounting Manager agents using the **lssvc** command. For example,
   ```
   $lssvc
   ITUAM_base
   ```
3. Based on the output of the **lssvc** command, decide which Tivoli Usage and Accounting Manager agent you want to configure. For example, `ITUAM_base`
4. List all of the attributes that are associated with the Tivoli Usage and Accounting Manager agent using the **cfgsvc** command. For example:
   ```
   $cfgsvc –ls ITUAM_base
    ACCT_DATA0
   ACCT_DATA1
   ISYSTEM
   IPROCESS
   ```
5. Configure the Tivoli Usage and Accounting Manager agent with its associated attributes using the **cfgsvc** command:
   ```
   cfgsvc ITUAM_agent_name -attr ACCT_DATA0=value1 ACCT_DATA1=value2 ISYSTEM=value3 IPROCESS=value4
   ```

   Where:
   - *ITUAM_agent_name* is the name of the Tivoli Usage and Accounting Manager agent. For example, ITUAM_base.
   - *value1* is the size (in MB) of the first data file that holds daily accounting information.
   - *value2* is the size (in MB) of the second data file that holds daily accounting information.
   - *value3* is the time (in minutes) when the agent generates system interval records.
   - *value4* is the time (in minutes) when the system generates aggregate process records.
6. Start the Tivoli Usage and Accounting Manager agent using the **startsvc** command. For example:
   ```
   startsvc ITUAM_base
   ```

After you start the Tivoli Usage and Accounting Manager agent, it begins to collect data and generate log files. You can configure the Tivoli Usage and Accounting Manager server to retrieve the log files, which are then processed by the Tivoli Usage and Accounting Manager Processing Engine. You can work with the data from the Tivoli Usage and Accounting Manager Processing Engine as follows:

- You can generate customized reports, spreadsheets, and graphs. Tivoli Usage and Accounting Manager provides full data access and reporting capabilities by integrating Microsoft® SQL Server Reporting Services or Crystal Reports with a Database Management System (DBMS).
- You can view high-level and detailed cost and usage information.
- You can allocate, distribute, or charge IT costs to users, cost centers, and organizations in a manner that is fair, understandable, and reproducible.

For more information, see the IBM Tivoli Usage and Accounting Manager Information Center.

Related reference

"Configuration attributes for IBM Tivoli agents and clients" on page 156
Learn about required and optional configuration attributes and variables for the IBM Tivoli Monitoring agent, the IBM Tivoli Usage and Accounting Manager agent, the IBM Tivoli Storage Manager client, and the IBM TotalStorage Productivity Center agents.

Configuring the IBM Tivoli Storage Manager client

You can configure theIBM Tivoli Storage Manager client on the Virtual I/O Server.

With Virtual I/O Server 1.4, you can configure the Tivoli Storage Manager client on the Virtual I/O Server. With Tivoli Storage Manager, you can protect your data from failures and other errors by storing backup and disaster-recovery data in a hierarchy of offline storage. Tivoli Storage Manager can help protect computers running a variety of different operating environments, including the Virtual I/O Server, on a variety of different hardware, including IBM System p servers. If you configure the Tivoli Storage Manager client on the Virtual I/O Server, you can include the Virtual I/O Server in your standard backup framework.

Before you start, ensure that the Virtual I/O Server is installed. The Tivoli Storage Manager client is packaged with the Virtual I/O Server and is installed when the Virtual I/O Server is installed. For instructions, see "Installing the Virtual I/O Server and client logical partitions" on page 78.

To configure and start the Tivoli Storage Manager client, complete the following steps:

1. List all of the available Tivoli Storage Manager clients using the **lssvc** command. For example,

   ```
   $lssvc
   TSM_base
   ```

2. Based on the output of the **lssvc** command, decide which Tivoli Storage Manager client you want to configure. For example, `TSM_base`
3. List all of the attributes that are associated with the Tivoli Storage Manager client using the **cfgsvc** command. For example:

   ```
   $cfgsvc –ls TSM_base
    SERVERNAME
   SERVERIP
   NODENAME
   ```

4. Configure the Tivoli Storage Manager client with its associated attributes using the **cfgsvc** command:

   ```
   cfgsvc TSM_client_name -attr SERVERNAME=hostname SERVERIP=name_or_address NODENAME=vios
   ```

   Where:

   - *TSM_client_name* is the name of the Tivoli Storage Manager client. For example, TSM_base.
   - *hostname* is the host name of the Tivoli Storage Manager server to which the Tivoli Storage Manager client is associated.

- *name_or_address* is the IP address or domain name of the Tivoli Storage Manager server to which the Tivoli Storage Manager client is associated.
- *vios* is the name of the machine on which the Tivoli Storage Manager client is installed. The name must match the name registered on the Tivoli Storage Manager server.

5. Ask the Tivoli Storage Manager administrator to register the client node, the Virtual I/O Server, with the Tivoli Storage Manager server. To determine what information you must provide to the Tivoli Storage Manager administrator, see the IBM Tivoli Storage Manager for UNIX and Linux Backup-Archive Clients Installation and User's Guide.

After you are finished, you are ready to back up and restore the Virtual I/O Server using the Tivoli Storage Manager. For instructions, see the following procedures:

- "Backing up the Virtual I/O Server using IBM Tivoli Storage Manager" on page 132
- "Restoring the Virtual I/O Server using IBM Tivoli Storage Manager" on page 137

Configuring the IBM TotalStorage Productivity Center agents

You can configure and start the IBM TotalStorage Productivity Center agents on the Virtual I/O Server.

With Virtual I/O Server 1.5.2, you can configure the IBM TotalStorage Productivity Center agents on the Virtual I/O Server. TotalStorage Productivity Center is an integrated, storage infrastructure management suite that is designed to help simplify and automate the management of storage devices, storage networks, and capacity utilization of file systems and databases. When you configure the TotalStorage Productivity Center agents on the Virtual I/O Server, you can use the TotalStorage Productivity Center user interface to collect and view information about the Virtual I/O Server.

Before you start, complete the following tasks:

1. Use the ioslevel command to verify that the Virtual I/O Server is at version 1.5.2, or later.
2. Ensure that there are no other operations running on the Virtual I/O Server. Configuring the TotalStorage Productivity Center consumes all of the processing time.
3. In addition to the memory required by the Virtual I/O Server logical partition, ensure that you have allocated a minimum of 1 GB of memory to the Virtual I/O Server for the TotalStorage Productivity Center agents.

To configure and start the TotalStorage Productivity Center agents, complete the following steps:

1. List all of the available TotalStorage Productivity Center agents using the **lssvc** command. For example,

   ```
   $lssvc
   TPC
   ```

   The TPC agent includes both the TPC_data and TPC_fabric agents. When you configure the TPC agent, you configure both the TPC_data and TPC_fabric agents.
2. List all of the attributes that are associated with the TotalStorage Productivity Center agent using the **lssvc** command. For example:

   ```
   $lssvc TPC
   A:
   S:
   devAuth:
   caPass:
   caPort:
   amRegPort:
   amPubPort:
   dataPort:
   devPort:
   newCA:
   oldCA:
   daScan:
   ```

```
daScript:
daInstall:
faInstall:
U:
```

The A, S, devAuth, and caPass attributes are required. The remainder of the attributes are optional. For more information about the attributes, see "Configuration attributes for IBM Tivoli agents and clients" on page 156.

3. Configure the TotalStorage Productivity Center agent with its associated attributes using the **cfgsvc** command:

   ```
   cfgsvc TPC -attr S=tpc_server_hostname A=agent_manager_hostname devAuth=password_1 caPass=password_2
   ```

   Where:
   - *tpc_server_hostname* is the host name or IP address of the TotalStorage Productivity Center server that is associated with the TotalStorage Productivity Center agent.
   - *agent_manager_hostname* is the name or IP address of the Agent Manager.
   - *password_1* is the password required to authenticate to the TotalStorage Productivity Center device server.
   - *password_2* is the password required to authenticate to the common agent.
4. Select the language that you want to use during the installation and configuration.
5. Accept the license agreement to install the agents according to the attributes specified in step 3.
6. Start each TotalStorage Productivity Center agent using the **startsvc** command:
   - To start the TPC_data agent, run the following command:

     ```
     startsvc TPC_data
     ```
   - To start the TPC_fabric agent, run the following command:

     ```
     startsvc TPC_fabric
     ```

After you start the TotalStorage Productivity Center agents, you can perform the following tasks using the TotalStorage Productivity Center user interface:

1. Run a discovery job for the agents on the Virtual I/O Server.
2. Run probes, scans, and ping jobs to collect storage information about the Virtual I/O Server.
3. Generate reports using the Fabric Manager and the Data Manager to view the storage information gathered.
4. View the storage information gathered using the topology Viewer.

For more information, see the *IBM TotalStorage Productivity Center support for agents on a Virtual I/O Server* PDF. To view or download the PDF, go to the IBM TotalStorage Productivity Center v3.3.1.81 Interim Fix Web site.

Configuring the IBM Director agent

You can configure and start the IBM Director agent on the Virtual I/O Server.

Before you start, use the ioslevel command to verify that the Virtual I/O Server is at version 1.5.2, or later.

With Virtual I/O Server 1.5.2, you can configure the IBM Director agent on the Virtual I/O Server. Using the IBM Director agent, you can view and track hardware configuration details of the system and monitor performance and use of critical components, such as processors, disks, and memory.

To configure and start the IBM Director agent, complete the following steps:

1. List the available IBM Director using the lssvc command. For example,

```
$lssvc
DIRECTOR_agent
```

2. Configure the IBM Director agent with its associated attributes using the cfgsvc command:

   ```
   cfgsvc DIRECTOR_agent -attr RESTART_ON_REBOOT=TRUE
   ```

   RESTART_ON_REBOOT designates whether the IBM Director agent restarts if the Virtual I/O Server is rebooted.
3. Start the IBM Director agent using the startsvc command. To start the DIRECTOR_agent agent, run the following command:

   ```
   startsvc DIRECTOR_agent
   ```

Related concepts

“IBM Systems Director software” on page 38
Learn about integrating the Virtual I/O Server into your IBM Systems Director environment.

Related information

cfgsvc command

Configuring the Virtual I/O Server as an LDAP client

Virtual I/O Server version 1.4 can be configured as an LDAP client and then you can manage Virtual I/O Server from an LDAP server.

Before you start, gather the following information:

- The name of the Lightweight Directory Access Protocol (LDAP) server or servers to which you want the Virtual I/O Server to be an LDAP client.
- The administrator distinguish name (DN) and password for the LDAP server or servers to which you want the Virtual I/O Server to be an LDAP client.

To configure the Virtual I/O Server as an LDAP client, complete the following steps:

1. Change Virtual I/O Server users to LDAP users by running the following command:

   ```
   chuser -ldap username
   ```

   where *username* is the name of the user you want to change to an LDAP user.
2. Set up the LDAP client by running the following command:

   ```
   mkldap –host ldapserv1 –bind cn=admin –passwd adminpwd
   ```

   Where:

   - *ldapserv1* is the LDAP server or list of LDAP servers to which you want the Virtual I/O Server to be an LDAP client
   - *cn=admin* is the administrator DN of *ldapserv1*
   - *adminpwd* is the password for *cn=admin*

   Configuring the LDAP client automatically starts communication between the LDAP server and the LDAP client (the Virtual I/O Server). To stop communication, use the stopnetsvc command.

Configuring the Virtual I/O Server for POWER6 systems

Before you upgrade an existing POWER5 Virtual I/O Server (VIOS) to a POWER6 VIOS, you must configure the maximum virtual I/O slot number and any virtual Ethernet, virtual serial, or virtual SCSI adapters that use VIOS slots 0 through 10.

The following configuration rules apply:

- The maximum virtual I/O slot number must be set to at least 11 plus the number of virtual I/O slots that you require.

Notes:

– A maximum that is lower than 11 can be incompatible with newer versions of the Hardware Management Console (HMC).

– The maximum slot number can be greater than 11.

– Excess virtual slots use a small amount of additional memory but have no other effects.

- All customer-defined virtual Ethernet, virtual serial, and virtual SCSI slots must use virtual slot IDs 11 or greater.

  Note: For existing virtual SCSI adapters, you must map all client profiles to the new server adapters.

These configuration rules apply to partitions on POWER6 systems only. In a mixture of POWER5 and POWER6 systems on a V7 HMC, the POWER5 systems can use slots 0 through 10.

Managing the Virtual I/O Server

You can manage virtual SCSI and virtual Ethernet devices on the Virtual I/O Server, as well as back up, restore, update, and monitor the Virtual I/O Server.

Most of the information in this topic is specific to management in an HMC environment. For information about management tasks in an Integrated Virtualization Manager environment, see Integrated Virtualization Manager.

Managing storage

You can import and export volume groups and storage pools, map virtual disks to physical disks, increase virtual SCSI device capacity, change the virtual SCSI queue depth, back up and restore files and file systems, and collect and view information using the IBM TotalStorage Productivity Center.

Importing and exporting volume groups and logical volume storage pools

You can use the importvg and exportvg commands to move a user-defined volume group from one system to another.

Consider the following when importing and exporting volume groups and logical volume storage pools:

- The import procedure introduces the volume group to its new system.
- You can use the importvg command to reintroduce a volume group or logical volume storage pool to the system that it had been previously associated with and had been exported from.
- The importvg command changes the name of an imported logical volume if a logical volume of that name already exists on the new system. If the importvg command must rename a logical volume, it prints an error message to standard error.
- The export procedure removes the definition of a volume group from a system.
- You can use the importvg and exportvg commands to add a physical volume that contains data to a volume group by putting the disk to be added in its own volume group.
- The rootvg volume group cannot be exported or imported.

Importing volume groups and logical volume storage pools:

You can use the importvg command to import a volume group or logical volume storage pool.

To import a volume group or logical volume storage pool, complete the following steps:

1. Run the following command to import the volume group or logical volume storage pool:

   ```
   importvg -vg volumeGroupName physicalVolumeName
   ```

   Where:

- *volumeGroupName* is an optional parameter that specifies the name to use for the imported volume group.
- *physicalVolumeName* is the name of a physical volume that belongs to the imported volume group.

2. If you know that the imported volume group or logical volume storage pool is not the parent of the virtual media repository or any file storage pools, then you are finished importing the volume group or logical volume storage pool and do not need to complete the remaining steps.
3. If you know that imported volume group or logical volume storage pool is the parent of the virtual media repository or any file storage pools, or if you are unsure, then complete the following steps:
   a. Run the `mount all` command to mount any file systems contained in the imported volume group or logical volume storage pool. This command might return errors for file systems that are already mounted.
   b. If you are importing a volume group or logical volume storage to the same system from which you exported it, run the `cfgdev` to reconfigure any devices that were unconfigured when you exported the volume group or logical volume storage pool.

To export a volume group or logical volume storage pool, see "Exporting volume groups and logical volume storage pools."

Exporting volume groups and logical volume storage pools:

You can use the exportvg command to export a volume group or logical volume storage pool.

Before you start, complete the following tasks:

1. Determine whether the volume group or logical volume storage pool that you plan to export is a parent to the virtual media repository or to any file storage pools by completing the following steps:
   a. Run the lsrep command to determine whether the volume group or logical volume storage pool that you plan to export is a parent of the virtual media repository. The Parent Pool field displays the parent volume group or logical volume pool of the virtual media repository.
   b. Run the following command to determine whether a file storage pool is a child of the volume group or logical volume pool that you plan to export:

      ```
      lssp -detail -sp FilePoolName
      ```

      The results list the parent volume group or logical volume storage pool of the file storage pool.
2. If the volume group or logical volume storage pool that you plan to export is a parent of the virtual media repository or a file storage pool, then complete the following steps.

Table 32. Prerequisites steps if the volume group or logical volume storage pool is a parent of the virtual media repository or a file storage pool

Parent of Virtual Media Repository	Parent of a file storage pool

1. Unload the backing device of each file-backed optical virtual target device (VTD) that has a media file loaded, by completing the following steps:
a. Retrieve a list of the file-backed optical VTDs by running the following command:
`lsmap -all -type file_opt`
b. For each device that shows a backing device, run the following command to unload the backing device:
`unloadopt -vtd` *`VirtualTargetDevice`*
2. Unmount the Virtual Media Repository file system by running the following command:
`unmount /var/vio/VMLibrary`
	1. Unconfigure the virtual target devices (VTDs) associated with the files contained in the file storage pools by completing the following steps:
	a. Retrieve a list of VTDs by running the following command:
	`lssp -bd -sp` *`FilePoolName`*
	where *FilePoolName* is the name of a file storage pool that is a child of the volume group or logical volume storage pool that you plan to export.
	b. For each file that lists a VTD, run the following command:
	`rmdev -dev` *`VirtualTargetDevice`* `-ucfg`
	2. Unmount the file storage pool by running the following command:
	`unmount /var/vio/storagepools/`*`FilePoolName`*
	where *FilePoolName* is the name of the file storage pool to be unmounted.

To export the volume group or logical volume storage pool, run the following commands:

1. `deactivatevg` *VolumeGroupName*
2. `exportvg` *VolumeGroupName*

To import a volume group or logical volume storage pool, see “Importing volume groups and logical volume storage pools” on page 117.

Mapping virtual disks to physical disks

Find instructions for mapping a virtual disk on a client logical partition to its physical disk on the Virtual I/O Server.

This procedure shows how to map a virtual SCSI disk on an AIX client logical partition to the physical device (disk or logical volume) on the Virtual I/O Server.

To map a virtual disk to a physical disk, you need the following information. This information is gathered during this procedure:

- Virtual device name
- Slot number of the virtual SCSI client adapter
- Logical unit number (LUN) of the virtual SCSI device
- Client logical partition ID

Follow these steps to map a virtual disk on an AIX client logical partition to its physical disk on the Virtual I/O Server:

1. Display virtual SCSI device information on the AIX client logical partition by typing the following command:

   ```
   lscfg -l devicename
   ```

   This command returns results similar to the following:

   ```
   U9117.570.1012A9F-V3-C2-T1-L810000000000  Virtual SCSI Disk Drive
   ```

2. Record the slot number, which is located in the output, following the card location label *C*. This identifies the slot number of the virtual SCSI client adapter. In this example, the slot number is 2.
3. Record the LUN, which is located in the output, following the LUN label *L*. In this example, the LUN is 810000000000.
4. Record the logical partition ID of the AIX client logical partition:
   a. Connect to the AIX client logical partition. For example, using Telnet.
   b. On the AIX logical partition, run the `uname -L` command.

      Your results should look similar to the following:

      ```
      2  fumi02
      ```

      The logical partition ID is the first number listed. In this example, the logical partition ID is 2. This number is used in the next step.
   c. Type `exit`.
5. If you have multiple Virtual I/O Server logical partitions running on your system, determine which Virtual I/O Server logical partition is serving the virtual SCSI device. Use the slot number of the client adapter that is linked to a Virtual I/O Server, and a server adapter. Use the HMC command line to list information about virtual SCSI client adapters in the client logical partition.

   Log in to the HMC, and from the HMC command line, type `lshwres` . Specify the managed console name for the **-m** parameter and the client logical partition ID for the **lpar_ids** parameter.

   Note:

   - The managed console name, which is used for the **-m** parameter, is determined by typing `lssyscfg -r sys -F name` from the HMC command line.
   - Use the client logical partition ID recorded in Step 4 for the **-lpar_ids** parameter.

   For example:

   ```
   lshwres -r virtualio --rsubtype scsi -m fumi --filter lpar_ids=2
   ```

   This example returns results similar to the following:

   ```
   lpar_name=fumi02,lpar_id=2,slot_num=2,state=null,adapter_type=client,remote_lpar_id=1,
   remote_lpar_name=fumi01,remote_slot_num=2,is_required=1,backing_devices=none
   ```

   Record the name of the Virtual I/O Server located in the **remote_lpar_name** field and slot number of the virtual SCSI server adapter, which is located in the **remote_lpar_id** field. In this example, the name of the Virtual I/O Server is fumi01 and the slot number of the virtual SCSI server adapter is 2.
6. Log in to the Virtual I/O Server.
7. List virtual adapters and devices on the Virtual I/O Server by typing the following command:

   ```
   lsmap -all
   ```
8. Find the virtual SCSI server adapter (vhost*X*) that has a slot ID that matches the remote slot ID recorded in Step 5. On that adapter, run the following command:

   ```
   lsmap -vadapter devicename
   ```
9. From the list of devices, match the LUN recorded in Step 3 with LUNs listed. This is the physical device.

Increasing virtual SCSI device capacity

Increase the size of virtual SCSI disks.

As storage demands increase for virtual client logical partitions, you can add physical storage to increase the size of your virtual devices and allocate that storage to your virtual environment. You can increase the capacity of your virtual SCSI devices by increasing the size of physical or logical volumes. With Virtual I/O Server version 1.3 and later, you can do this without disrupting client operations. To increase the size of files and logical volumes based on storage pools, the Virtual I/O Server must be at version 1.5 or later. To update the Virtual I/O Server, see “Updating the Virtual I/O Server” on page 126.

Tip: If you are using the HMC, version 7 release 3.4.2 or later, you can use the HMC graphical interface to increase the capacity of a virtual SCSI device on aVirtual I/O Server.

To increase virtual SCSI device capacity, complete the following steps:

1. Increase the size of the physical volumes, logical volumes, or files:
   - Physical volumes: Consult your storage documentation to determine whether your storage subsystem supports expanding the size of a logical unit number (LUN).
   - Logical volumes based on volume groups:
     a. Run the extendlv command. For example: `extendlv lv3 100M`. This example increases logical volume *lv3* by 100 MB.
     b. If there is no additional space in the logical volume, complete the following tasks:
        1) Increase the size of the volume group by completing one of the following steps:
           – Increase the size of the physical volumes. Consult your storage documentation for instructions.
           – Add physical volumes to a volume group by running the extendvg command. For example: `extendvg vg1 hdisk2`. This example adds physical volume *hdisk2* to volume group *vg1*.
        2) Allocate the increased volume to partitions by resizing logical volumes. Run the extendlv command to increase the size of a logical volume.
   - Logical volumes based on storage pools:
     a. Run the chbdsp command. For example:`chbdsp -sp lvPool -bd lv3 -size 100M`. This example increases logical volume *lv3* by 100 MB.
     b. If there is no additional space in the logical volume, complete the following tasks:
        1) Increase the size of the logical volume storage pool by completing one of the following steps:
           – Increase the size of the physical volumes. Consult your storage documentation for instructions.
           – Add physical volumes to the storage pool by running the chsp command. For example: `chsp -add -sp sp1 hdisk2`. This example adds physical volume *hdisk2* to storage pool *sp1*.
        2) Allocate the increased volume to partitions by resizing logical volumes. Run the chbdsp command to increase the size of a logical volume.
   - Files:
     a. Run the chbdsp command. For example:`chbdsp -sp fbPool -bd fb3 -size 100M`. This example increases file *fb3* by 100 MB.
     b. If there is no additional space in the file, increase the size of the file storage pool by running the chsp command. For example:`chsp -add -sp fbPool -size 100M`. This example increases file storage pool *fbPool* by 100MB.
     c. If there is no additional space in the file storage pool, increase the size of the parent storage pool by completing one of the following tasks:
        – Increase the size of the physical volumes. Consult your storage documentation for instructions.
        – Add physical volumes to the parent storage pool by running the chsp command. For example:`chsp -add -sp sp1 hdisk2`. This example adds physical volume *hdisk2* to storage pool *sp1*.
        – Increase the size of the file storage pool by running the chsp command.
2. If you are running Virtual I/O Server versions prior to 1.3, then you need to either reconfigure the virtual device (using the cfgdev command) or restart the Virtual I/O Server.

3. If you are running Virtual I/O Server version 1.3 or later, then restarting or reconfiguring a logical partition is not required to begin using the additional resources. If the physical storage resources have been set up and properly allocated to the system as a system resource, as soon as the Virtual I/O Server recognizes the changes in storage volume, the increased storage capacity is available to the client logical partitions.
4. On the client logical partition, ensure that the operating system recognizes and adjusts to the new size. For example, if AIX is the operating system on the client logical partition, run the following command:

   ```
   chvg -g vg1
   ```

   In this example, AIX examines all the disks in volume group *vg1* to see if they have grown in size. For the disks that have grown in size, AIX attempts to add additional physical partitions to physical volumes. If necessary, AIX will determine proper 1016 multiplier and conversion to the big volume group.

Related information

chvg Command

IBM System p Advanced POWER Virtualization Best Practices RedPaper

Changing a storage pool for a VIOS logical partition using the HMC

Changing the virtual SCSI queue depth

Increasing the virtual SCSI queue depth might provide performance improvements for some virtual configurations. Understand the factors involved in determining a change to the virtual SCSI queue depth value.

The virtual SCSI queue depth value determines how many requests the disk head driver will queue to the virtual SCSI client driver at any one time. For AIX and Linux client logical partitions, you can change this value from the default value of 3 to any value from 1 to 256. You modify this value using the chdev command. For IBM i client logical partitions, the queue depth value is 32 and cannot be changed.

Increasing this value might improve the throughput of the disk in specific configurations. However, several factors must be taken into consideration. These factors include the value of the queue-depth attribute for all of the physical storage devices on the Virtual I/O Server being used as a virtual target device by the disk instance on the client logical partition, and the maximum transfer size for the virtual SCSI client adapter instance that is the parent device for the disk instance.

For AIX and Linux client logical partitions, the maximum transfer size for virtual SCSI client adapters is set by the Virtual I/O Server, which determines the value based on the resources available on the server and the maximum transfer size set for the physical storage devices on that server. Other factors include the queue depth and maximum transfer size of other devices involved in mirrored-volume-group or Multipath I/O (MPIO) configurations. Increasing the queue depth for some devices might reduce the resources available for other devices on that same shared adapter and decrease the throughput for those devices. For IBM i client logical partitions, the queue depth value is 32 and cannot be changed.

To change the queue depth for an AIX or Linux client logical partition, on the client logical partition use the chdev command with the **queue_depth=value** attribute as in the following example:

```
chdev -1 hdiskN -a "queue_depth=value"
```

hdiskN represents the name of a physical volume and *value* is the value you assign between 1 and 256.

To view the current setting for the queue_depth value, from the client logical partition issue the following command:

```
lsattr -El hdiskN
```

Backing up and restoring files and file systems

You can use the backup and restore commands to back up and restore individual files or entire file systems.

Backing up and restoring files and files systems can be useful for tasks, such as saving IBM i to physical tape or saving a file-backed device.

The following commands are used to back up and restore files and files systems.

Table 33. Backup and restore commands and their descriptions

Command	Description
backup	Backs up files and file systems to media, such as physical tape and disk. For example: • You can back up all the files and subdirectories in a directory using full path names or relative path names. • You can back up the root file system. • You can back up all the files in the root file system that have been modified since the last backup. • You can back up virtual optical media files from the virtual media repository.
restore	Reads archives created by the backup command and extracts the files stored there. For example: • You can restore a specific file into the current directory. • You can restore a specific file from tape into the virtual media repository. • You can restore a specific directory and the contents of that directory from a file name archive or a file system archive. • You can restore an entire file system. • You can restore only the permissions or only the ACL attributes of the files from the archive.

Managing storage using the IBM TotalStorage Productivity Center

You can use the IBM TotalStorage Productivity Center collect and view information about the Virtual I/O Server.

With Virtual I/O Server 1.5.2, you can install and configure the TotalStorage Productivity Center agents on the Virtual I/O Server. TotalStorage Productivity Center is an integrated, infrastructure management suite for storage that is designed to help simplify and automate the management of storage devices, storage networks, and capacity utilization of file systems and databases. When you install and configure the TotalStorage Productivity Center agents on the Virtual I/O Server, you can use the TotalStorage Productivity Center interface to collect and view information about the Virtual I/O Server. You can then perform the following tasks using the TotalStorage Productivity Center interface:

1. Run a discovery job for the agents on the Virtual I/O Server.
2. Run probes, run scans, and ping jobs to collect storage information about the Virtual I/O Server.
3. Generate reports using the Fabric Manager and the Data Manager to view the storage information gathered.
4. View the storage information gathered using the topology Viewer.

For more information, see “Configuring the IBM TotalStorage Productivity Center agents” on page 114.

Managing networks

You can change the network configuration of the Virtual I/O Server logical partition, enable and disable GARP VLAN Registration Protocol (GVRP) on your Shared Ethernet Adapters, use Simple Network Management Protocol (SNMP) to manage systems and devices in complex networks, and upgrade to Internet Protocol version 6 (IPv6).

Changing the network configuration of the Virtual I/O Server logical partition

Follow these steps to change or remove the network settings on the Virtual I/O Server logical partition, such as the IP address, subnet mask, gateway, and nameserver address

In this scenario, the Virtual I/O Server logical partition already has its network configuration set. The current configuration will be removed, and the updated configuration will then be set. If you plan to undo your Internet Protocol version 6 (IPv6) configuration, use the following process and commands to completely remove the TCP/IP interface and then configure a new TCP/IP interface for Internet Protocol version 4 (IPv4).

1. View the current network configuration using the lstcpip command.
2. Remove the current network configuration by running the rmtcpip command. You can remove all network settings or just the specific settings that need to be updated.
3. Configure the new network settings using the mktcpip command.

The following example is for IPv4 where the Virtual I/O Server logical partition needs to have its domain name server (DNS) information updated from its current address to 9.41.88.180:

1. Run `lstcpip -namesrv` to view the current configuration. Ensure you want to update this configuration.
2. Run `rmtcpip -namesrv` to remove the current configuration.
3. Run `mktcpip -nsrvaddr 9.41.88.180` to update the nameserver address.

Enabling and disabling GVRP

You can enable and disable GARP VLAN Registration Protocol (GVRP) on your Shared Ethernet Adapters to control dynamic registration of VLANs over networks.

With Virtual I/O Server version 1.4, Shared Ethernet Adapters support GARP VLAN Registration Protocol (GVRP) which is based on GARP (Generic Attribute Registration Protocol). GVRP allows for the dynamic registration of VLANs over networks.

By default, GVRP is disabled on Shared Ethernet Adapters.

Before you start, create and configure the Shared Ethernet Adapter. For instructions, see “Creating a virtual Ethernet adapter using HMC version 7” on page 104.

To enable or disable GVRP, run the following command:

```
chdev -dev Name -attr gvrp=yes/no
```

Where:

- *Name* is the name of the Shared Ethernet Adapter.
- *yes/no* defines whether GVRP is enabled or disabled. Type `yes` to enable GVRP and type `no` to disable GVRP.

Managing SNMP on the Virtual I/O Server

Find commands for enabling, disabling, and working with SNMP on the Virtual I/O Server.

Simple Network Management Protocol (SNMP) is a set of protocols for monitoring systems and devices in complex networks. SNMP network management is based on the familiar client-server model that is widely used in Internet protocol (IP) network applications. Each managed host runs a process called an agent. The agent is a server process that maintains information about managed devices in the Management Information Base (MIB) database for the host. Hosts that are involved in network management decision-making can run a process called a manager. A manager is a client application that generates requests for MIB information and processes responses. In addition, a manager might send requests to agent servers to modify MIB information.

In general, network administrators use SNMP to more easily manage their networks for the following reasons:

- It hides the underlying system network
- The administrator can manage and monitor all network components from one console

SNMP is available on Virtual I/O Server version 1.4 and later.

The following table lists the SNMP management tasks available on the Virtual I/O Server, as well as the commands you need to run to accomplish each task.

Table 34. Tasks and associated commands for working with SNMP on the Virtual I/O Server

Task	**Command**
Enable SNMP	startnetsvc
Select which SNMP agent you want to run	snmpv3_ssw
Issue SNMP requests to agents	cl_snmp
Process SNMP responses returned by agents	cl_snmp
Request MIB information managed by an SNMP agent	snmp_info
Modify MIB information managed by an SNMP agent	snmp_info
Generate a notification, or trap, that reports an event to the SNMP manager with a specified message	snmp_trap
Disable SNMP	stopnetsvc

Related information

Network Management

Upgrading the Virtual I/O Server from IPv4 to IPv6

To take advantage of enhancements, such as expanded addressing and routing simplification, use the mktcpip command to upgrade the Virtual I/O Server from Internet Protocol version 4 (IPv4) to Internet Protocol version 6 (IPv6).

IPv6 is the next generation of Internet protocol and is gradually replacing the current Internet standard, Internet Protocol version 4 (IPv4). The key IPv6 enhancement is the expansion of the IP address space from 32 bits to 128 bits, providing virtually unlimited, unique IP addresses. IPv6 provides several advantages over IPv4 including expanded routing and addressing, routing simplification, header format simplification, improved traffic control, autoconfiguration, and security.

Run the following command to upgrade from the Virtual I/O Server from IPv4 to IPv6:

```
mktcpip –auto [-interface interface]
```

where *interface* specifies which interface you want to configure for IPv6.
This command automatically performs the following tasks:

- Configures all link-local addresses for IPv6 that are currently configured for IPv4.
- Turns on the specified interfaces daemonthat support IPv6.
- Starts the ndpd-host daemon.
- Ensures that the IPv6 configuration remains intact after you reboot the Virtual I/O Server.

If you decide that you want to undo the IPv6 configuration, you must completely remove the TCP/IP interface and then configure a new TCP/IP interface for IPv4. For instructions, see “Changing the network configuration of the Virtual I/O Server logical partition” on page 124.

Subscribe to Virtual I/O Server product updates

A subscription service is available to allow Virtual I/O Server users to stay current on news and product updates.

To subscribe to this service, follow these steps:

1. Go to the Subscription service for UNIX and Linux servers Web site.
2. Click the **Subscribe / Setup** tab and complete the form.

After subscribing, you are notified of all Virtual I/O Server news and product updates.

Updating the Virtual I/O Server

To install an update to the Virtual I/O Server, you can obtain the update either from a CD that contains the update or download the update.

To update the Virtual I/O Server, follow these steps:

1. Make a backup of the Virtual I/O Server by following the steps in Backing up the Virtual I/O Server.
2. Download the required updates from the Virtual I/O Server support site. Alternatively, you can get the updates from the update CD.
3. Install the update using the updateios command. For example, if your update fileset is located in the /home/padmin/update directory, type the following:

   ```
   updateios -install -accept -dev /home/padmin/update
   ```

   Note: The updateios command installs all updates located in the specified directory.

Backing up the Virtual I/O Server

You can back up the Virtual I/O Server and user-defined virtual devices using the backupios command. You can also use IBM Tivoli Storage Manager to schedule backups and store backups on another server.

The Virtual I/O Server contains the following types of information that you need to back up: the Virtual I/O Server itself and user-defined virtual devices.

- The Virtual I/O Server includes the base code, applied fix packs, custom device drivers to support disk subsystems, and some user-defined metadata. All of this information is backed up when you use the backupios command. In situations where you plan to restore the Virtual I/O Server to the same system from which it was backed up, then backing up only the Virtual I/O Server itself is usually sufficient.
- User-defined virtual devices include metadata, like virtual devices mappings, that define the relationship between the physical environment and the virtual environment. This data can be saved to a location that is automatically backed up when you use the backupios command. In situations where you plan to restore the Virtual I/O Server to a new or different system (for example, in the event of a system failure or disaster), then you must back up both the Virtual I/O Server and user-defined virtual devices. Furthermore, in these situations, you must also back up the following components of your environment in order to fully recover your Virtual I/O Server configuration:
  - External device configurations, such as Storage Area Network (SAN) devices.
  - Resources defined on the Hardware Management Console (HMC), such as processor and memory allocations. This means backing up your HMC partition profile data for the Virtual I/O Server and its client partitions.
  - The operating systems and applications running in the client logical partitions.

You can back up and restore the Virtual I/O Server as follows.

Table 35. Backup and restoration methods for the Virtual I/O Server

Backup method	Media	Restoration method
To tape	Tape	From tape
To DVD	DVD-RAM	From DVD
To remote file system	nim_resources.tar image	From an HMC using the Network Installation Management (NIM) on Linux facility and the installios command
To remote file system	mksysb image	From an AIX 5L™ NIM server and a standard mksysb system installation
Tivoli Storage Manager	mksysb image	Tivoli Storage Manager

Backing up the Virtual I/O Server to tape

You can back up the Virtual I/O Server base code, applied fix packs, custom device drivers to support disk subsystems, and some user-defined metadata to tape.

If the system is managed by the Integrated Virtualization Manager, then you need to back up your partition profile data for the management partition and its clients before you back up the Virtual I/O Server. For instructions, see Backing up and restoring partition data. (Alternatively, you can use the bkprofdata command.)

To back up the Virtual I/O Server to tape, follow these steps:

1. Assign a tape drive to the Virtual I/O Server.
2. Get the device name by typing the following command:
   ```
   lsdev -type tape
   ```
   If the tape device is in the `Defined` state, type the following command, where *dev* is the name of your tape device:
   ```
   cfgdev -dev dev
   ```
3. Type the following command, where *tape_device* is the name of the tape device you want to back up to:
   ```
   backupios -tape tape_device
   ```
   This command creates a bootable tape that you can use to restore the Virtual I/O Server.
4. If you plan to restore the Virtual I/O Server to a different system from which it was backed up, then you need to back up the user-defined virtual devices. For instructions, see "Backing up user-defined virtual devices" on page 130.

Related information

IBM System p Advanced POWER Virtualization Best Practices RedPaper

Backing up the Virtual I/O Server to one or more DVDs

You can back up the Virtual I/O Server base code, applied fix packs, custom device drivers to support disk subsystems, and some user-defined metadata to DVD.

If the system is managed by the Integrated Virtualization Manager, then you need to back up your partition profile data for the management partition and its clients before you back up the Virtual I/O Server. For instructions, see Backing up and restoring partition data. (Alternatively, you can use the bkprofdata command.)

To back up the Virtual I/O Server to one or more DVDs, follow these steps. Only DVD-RAM media can be used to back up the Virtual I/O Server.

Note: Vendor disk drives might support burning to additional disk types, such as CD-RW and DVD-R. Refer to the documentation for your drive to determine which disk types are supported.

1. Assign an optical drive to the Virtual I/O Server logical partition.
2. Get the device name by typing the following command:

   ```
   lsdev -type optical
   ```

   If the device is in the `Defined` state, type:

   ```
   cfgdev -dev dev
   ```

3. Run the backupios command with the **-cd** option. Specify the path to the device. For example:

   ```
   backupios -cd /dev/cd0
   ```

   Note: If the Virtual I/O Server does not fit on one DVD, then the backupios command provides instructions for disk replacement and removal until all the volumes have been created.
   This command creates one or more bootable DVDs that you can use to restore the Virtual I/O Server.
4. If you plan to restore the Virtual I/O Server to a different system from which it was backed up, then you need to back up the user-defined virtual devices. For instructions, see "Backing up user-defined virtual devices" on page 130.

Related information

IBM System p Advanced POWER Virtualization Best Practices RedPaper

Backing up the Virtual I/O Server to a remote file system by creating a nim_resources.tar file

You can back up the Virtual I/O Server base code, applied fix packs, custom device drivers to support disk subsystems, and some user-defined metadata to a remote file system by creating a nim_resources.tar file.

Backing up the Virtual I/O Server to a remote file system will create the nim_resources.tar image in the directory you specify. The nim_resources.tar file contains all the necessary resources to restore the Virtual I/O Server, including the mksysb image, the bosinst.data file, the network boot image, and Shared Product Object Tree (SPOT) resource.

The backupios command empties the target_disks_stanza section of bosinst.data and sets `RECOVER_DEVICES=Default`. This allows the mksysb file generated by the command to be cloned to another logical partition. If you plan to use the nim_resources.tar image to install to a specific disk, then you need to repopulate the target_disk_stanza section of bosinst.data and replace this file in the nim_resources.tar image. All other parts of the nim_resources.tar image must remain unchanged.

Before you start, complete the following tasks:

1. If the system is managed by the Integrated Virtualization Manager, then you need to back up your partition profile data for the management partition and its clients before you back up the Virtual I/O Server. For instructions, see Backing up and restoring partition data. (Alternatively, you can use the bkprofdata command.)
2. Ensure that the remote file system is available and mounted.
3. Ensure that the Virtual I/O Server has root write access to the server on which the backup will be created.

To back up the Virtual I/O Server to a remote file system, follow these steps:

1. Create a mount directory where the backup image, nim_resources.tar, will be written. For example, to create the directory /home/backup, type:

   ```
   mkdir /home/backup
   ```

2. Mount an exported directory on the mount directory. For example:

   ```
   mount server1:/export/ios_backup /home/backup
   ```

3. Run the **backupios** command with the **-file** option. Specify the path to the mounted directory. For example:

   ```
   backupios -file /home/backup
   ```

   This command creates a nim_resources.tar file that you can use to restore the Virtual I/O Server from the HMC.

4. If you plan to restore the Virtual I/O Server to a different system from which it was backed up, then you need to back up the user-defined virtual devices. For instructions, see "Backing up user-defined virtual devices" on page 130.

Related information

IBM System p Advanced POWER Virtualization Best Practices RedPaper

Backing up the Virtual I/O Server to a remote file system by creating a mksysb image

You can back up the Virtual I/O Server base code, applied fix packs, custom device drivers to support disk subsystems, and some user-defined metadata to a remote file system by creating a mksysb file.

Backing up the Virtual I/O Server to a remote file system will create the mksysb image in the directory you specify. The mksysb image is an installable image of the root volume group in a file.

Before you start, complete the following tasks:

1. If the system is managed by the Integrated Virtualization Manager, then you need to back up your partition profile data for the management partition and its clients before you back up the Virtual I/O Server. For instructions, see Backing up and restoring partition data. (Alternatively, you can use the bkprofdata command.)
2. If you plan to restore the Virtual I/O Server from a Network Installation Management (NIM) server, verify that the NIM server is at the latest release of AIX. To find the latest updates, see the Fix Central Web site.
3. Ensure that the remote file system is available and mounted.
4. Ensure that the Virtual I/O Server has root write access to the server on which the backup will be created.

To back up the Virtual I/O Server to a remote file system, follow these steps:

1. Create a mount directory where the backup image, mksysb image, will be written. For example, to create the directory /home/backup, type:

   ```
   mkdir /home/backup
   ```

2. Mount an exported directory on the mount directory. For example:

   ```
   mount server1:/export/ios_backup /home/backup
   ```

   where *server1* is the NIM server from which you plan to restore the Virtual I/O Server.

3. Run the backupios command with the **-file** option. Specify the path to the mounted directory. For example:

   ```
   backupios -file /home/backup/filename.mksysb -mksysb
   ```

   where *filename* is the name of mksysb image that this command creates in the specified directory. You can use the mksysb image to restore the Virtual I/O Server from a NIM server.

4. If you plan to restore the Virtual I/O Server to a different system from which it was backed up, then you need to back up the user-defined virtual devices. For instructions, see "Backing up user-defined virtual devices" on page 130.

Backing up user-defined virtual devices

In addition to backing up the Virtual I/O Server, you need to back up user-defined virtual devices (such as virtual device mappings) in preparation of a system failure or disaster.

User-defined virtual devices include metadata, such as virtual device mappings, that define the relationship between the physical environment and the virtual environment. In situations where you plan to restore the Virtual I/O Server to a new or different system (for example, in the event of a system failure or disaster), you need to back up both the Virtual I/O Server and user-defined virtual devices.

Before you start, complete the following tasks:

1. Back up the Virtual I/O Server to tape, DVD, or a remote file system. For instructions, see one of the following procedures:
   - "Backing up the Virtual I/O Server to tape" on page 127
   - "Backing up the Virtual I/O Server to one or more DVDs" on page 127
   - "Backing up the Virtual I/O Server to a remote file system by creating a nim_resources.tar file" on page 128
   - "Backing up the Virtual I/O Server to a remote file system by creating a mksysb image" on page 129
2. Decide whether you want to create a script of the following procedure. Scripting these commands makes it easy to schedule automated backups of the information.

To back up user-defined virtual devices, complete the following steps:

1. List volume groups (and storage pools) to determine what user-defined disk structures you want to back up by running the following command:
   ```
   lsvg
   ```
2. Activate each volume group (and storage pool) that you want to back up by running the following command for each volume group:
   ```
   activatevg volume_group
   ```
   where *volume_group* is the name of the volume group (or storage pool) that you want to activate.
3. Back up each volume group (and storage pool) by running the following command for each volume group:
   ```
   savevgstruct volume_group
   ```
   where *volume_group* is the name of the volume group (or storage pool) that you want to back up. This command writes a backup of the structure of a volume group (and therefore a storage pool) to the **/home/ios/vgbackups** directory.
4. Save the information about network settings, adapters, users, and security settings to the /home/padmin directory by running each command in conjunction with the tee command as follows:
   ```
   command | tee /home/padmin/filename
   ```
   Where:
   - *command* is the command that produces the information you want to save.
   - *filename* is the name of the file to which you want to save the information.

Table 36. Commands that provide the information to save

Command	Information provided
cfgnamesrv -ls	Shows all system configuration database entries related to domain name server information used by local resolver routines.

Table 36. Commands that provide the information to save (continued)

Command	Information provided
`entstat -all devicename` *devicename* is the name of a device whose attributes or statistics you want to save. Run this command for each device whose attributes or statistics you want to save.	Shows Ethernet driver and device statistics for the device specified.
`hostmap -ls`	Shows all entries in the system configuration database.
`ioslevel`	Shows the current maintenance level of the Virtual I/O Server.
`lsdev -dev devicename -attr` *devicename* is the name of a device whose attributes or statistics you want to save. Run this command for each device whose attributes or statistics you want to save.	Shows the attributes of the device specified.
`lsdev -type adapter`	Shows information about physical and logical adapters.
`lsuser`	Shows a list of all attributes of all the system users.
`netstat -routinfo`	Shows the routing tables, including the user-configured and current costs of each route.
`netstat -state`	Shows the state of all configured interfaces.
`optimizenet -list`	Shows characteristics of all network tuning parameters, including the current and reboot value, range, unit, type, and dependencies.
`viosecure -firewall view`	Shows a list of allowed ports.
`viosecure -view -nonint`	Shows all of the security level settings for noninteractive mode.

Related information

IBM System p Advanced POWER Virtualization Best Practices RedPaper

Scheduling backups of the Virtual I/O Server

You can schedule regular backups of the Virtual I/O Server and user-defined virtual devices to ensure that your backup copy accurately reflects the current configuration.

To ensure that your backup of the Virtual I/O Server accurately reflects your current running Virtual I/O Server, you should back up the Virtual I/O Server each time that its configuration changes. For example:

- Changing the Virtual I/O Server, like installing a fix pack.
- Adding, deleting, or changing the external device configuration, like changing the SAN configuration.
- Adding, deleting, or changing resource allocations and assignments for the Virtual I/O Server, like memory, processors, or virtual and physical devices.
- Adding, deleting, or changing user-defined virtual device configurations, like virtual device mappings.

Before you start, ensure that you are logged into the Virtual I/O Server as the prime administrator (padmin).

To back up the Virtual I/O Server and user-defined virtual devices, complete the following tasks:

1. Create a script for backing up the Virtual I/O Server, and save it in a directory that is accessible to the **padmin** user ID. For example, create a script called *backup* and save it in the `/home/padmin` directory. Ensure that your script includes commands for backing up the Virtual I/O Server and saving information about user-defined virtual devices.

2. Create a **crontab** file entry that runs the *backup* script on a regular interval. For example, to run *backup* every Saturday at 2:00 a.m., type the following commands:

   a. `crontab -e`

   b. `0 2 0 0 6 /home/padmin/backup`

   When you are finished, remember to save and exit.

Related information

IBM System p Advanced POWER Virtualization Best Practices RedPaper

Backing up the Virtual I/O Server using IBM Tivoli Storage Manager

You can use the IBM Tivoli Storage Manager to automatically back up the Virtual I/O Server on regular intervals, or you can perform incremental backups.

Backing up the Virtual I/O Server using IBM Tivoli Storage Manager automated backup:

You can automate backups of the Virtual I/O Server using the crontab command and the IBM Tivoli Storage Manager scheduler.

Before you start, complete the following tasks:

- Ensure that you configured the Tivoli Storage Manager client on the Virtual I/O Server. For instructions, see "Configuring the IBM Tivoli Storage Manager client" on page 113.
- Ensure that you are logged into the Virtual I/O Server as the prime administrator (padmin).

To automate backups of the Virtual I/O Server, complete the following steps:

1. Write a script that creates a mksysb image of the Virtual I/O Server and save it in a directory that is accessible to the **padmin** user ID. For example, create a script called *backup* and save it in the `/home/padmin` directory. If you plan to restore the Virtual I/O Server to a different system from which it was backed up, then ensure that your script includes commands for saving information about user-defined virtual devices. For more information, see the following tasks:
   - For instructions about how to create a mksysb image, see "Backing up the Virtual I/O Server to a remote file system by creating a mksysb image" on page 129.
   - For instructions about how to save user-defined virtual devices, see "Backing up user-defined virtual devices" on page 130.
2. Create a crontab file entry that runs the *backup* script on a regular interval. For example, to create a mksysb image every Saturday at 2:00 a.m., type the following commands:

   a. `crontab -e`

   b. `0 2 0 0 6 /home/padmin/backup`

   When you are finished, remember to save and exit.
3. Work with the Tivoli Storage Manager administrator to associate the Tivoli Storage Manager client node with one or more schedules that are part of the policy domain. This task is not performed on the Tivoli Storage Manager client on the Virtual I/O Server. This task is performed by the Tivoli Storage Manager administrator on the Tivoli Storage Manager server.
4. Start the client scheduler and connect to the server schedule using the dsmc command as follows:

   ```
   dsmc -schedule
   ```
5. If you want the client scheduler to restart when the Virtual I/O Server restarts, then add the following entry to the /etc/inittab file:

   ```
   itsm::once:/usr/bin/dsmc sched > /dev/null 2>&1 # TSM scheduler
   ```

Related information

IBM Tivoli Storage Manager for UNIX and Linux Backup-Archive Clients Installation and User’s Guide

Backing up the Virtual I/O Server using IBM Tivoli Storage Manager incremental backup:

You can back up the Virtual I/O Server at any time by performing an incremental backup with the IBM Tivoli Storage Manager.

Perform incremental backups in situations where the automated backup does not suit your needs. For example, before you upgrade the Virtual I/O Server, perform an incremental backup to ensure that you have a backup of the current configuration. Then, after you upgrade the Virtual I/O Server, perform another incremental backup to ensure that you have a backup of the upgraded configuration.

Before you start, complete the following tasks:

- Ensure that you configured the Tivoli Storage Manager client on the Virtual I/O Server. For instructions, see “Configuring the IBM Tivoli Storage Manager client” on page 113.
- Ensure that you have a mksysb image of the Virtual I/O Server. If you plan to restore the Virtual I/O Server to a different system from which it was backed up, then ensure that the mksysb includes information about user-defined virtual devices. For more information, see the following tasks:
  - For instructions about how to create a mksysb image, see “Backing up the Virtual I/O Server to a remote file system by creating a mksysb image” on page 129.
  - For instructions about how to save user-defined virtual devices, see “Backing up user-defined virtual devices” on page 130.

To perform an incremental backup of the of the Virtual I/O Server, run the dsmc command. For example,

```
dsmc -incremental sourcefilespec
```

Where *sourcefilespec* is the directory path to where the mksysb file is located. For example, `/home/padmin/mksysb_image`.

Related information

IBM Tivoli Storage Manager for UNIX and Linux Backup-Archive Clients Installation and User’s Guide

Restoring the Virtual I/O Server

You can restore the Virtual I/O Server and user-defined virtual devices using the installios command or IBM Tivoli Storage Manager.

The Virtual I/O Server contains the following types of information that you need to restore: the Virtual I/O Server itself and user-defined virtual devices.

- The Virtual I/O Server includes the base code, applied fix packs, custom device drivers to support disk subsystems, and some user-defined metadata. All of this information is restored when you use the installios command. In situations where you restore the Virtual I/O Server to the same system on which it was backed up, then restoring only the Virtual I/O Server itself is usually sufficient.
- User-defined virtual devices include metadata, such as virtual devices mappings, that define the relationship between the physical environment and the virtual environment. You can use this data to recreate the virtual devices. In situations where you restore the Virtual I/O Server to a new or different system (for example, in the event of a system failure or disaster), then you need to restore the Virtual I/O Server and recreate the virtual devices. Furthermore, in these situations, you also need to restore the following components of your environment in order to fully recover your Virtual I/O Server configuration:
  - External device configurations, such as Storage Area Network (SAN) devices.

- Resources defined on the Hardware Management Console (HMC), such as processor and memory allocations. This means restoring your HMC partition profile data for the Virtual I/O Server and its client partitions.
- The operating systems and applications running in the client logical partitions.

You can back up and restore the Virtual I/O Server as follows.

Table 37. Backup and restoration methods for the Virtual I/O Server

Backup method	Media	Restoration method
To tape	Tape	From tape
To DVD	DVD-RAM	From DVD
To remote file system	nim_resources.tar image	From an HMC using the Network Installation Management (NIM) on Linux facility and the installios command
To remote file system	mksysb image	From an AIX 5L NIM server and a standard mksysb system installation
Tivoli Storage Manager	mksysb image	Tivoli Storage Manager

Restoring the Virtual I/O Server from tape

You can restore the Virtual I/O Server base code, applied fix packs, custom device drivers to support disk subsystems, and some user-defined metadata from tape.

If the system is managed by the Integrated Virtualization Manager, then you need to restore your partition profile data for the management partition and its clients before you restore the Virtual I/O Server. For instructions, see Backing up and restoring partition data. (Alternatively, you can use the rstprofdata command.)

To restore the Virtual I/O Server from tape, follow these steps:

1. Specify the Virtual I/O Server logical partition to boot from the tape by using the bootlist command. Alternatively, you can alter the bootlist in the System Management Services (SMS).
2. Insert the tape into the tape drive.
3. From the SMS menu, select to install from the tape drive.
4. Follow the installation steps according to the system prompts.
5. If you restored the Virtual I/O Server to a different system from which it was backed up, then you need to restore the user-defined virtual devices. For instructions, see “Restoring user-defined virtual devices” on page 136.

Related information

IBM System p Advanced POWER Virtualization Best Practices RedPaper

Restoring the Virtual I/O Server from one or more DVDs

You can restore the Virtual I/O Server base code, applied fix packs, custom device drivers to support disk subsystems, and some user-defined metadata from one or more DVDs.

If the system is managed by the Integrated Virtualization Manager, then you need to restore your partition profile data for the management partition and its clients before you restore the Virtual I/O Server. For instructions, see Backing up and restoring partition data. (Alternatively, you can use the rstprofdata command.)

To restore the Virtual I/O Server from a one or more DVDs, follow these steps:

1. Specify the Virtual I/O Server partition to boot from the DVD by using the **bootlist** command. Alternatively, you can alter the bootlist in the System Management Services (SMS).
2. Insert the DVD into the optical drive.
3. From the SMS menu, select to install from the optical drive.
4. Follow the installation steps according to the system prompts.
5. If you restored the Virtual I/O Server to a different system from which it was backed up, then you need to restore the user-defined virtual devices. For instructions, see "Restoring user-defined virtual devices" on page 136.

Related information

IBM System p Advanced POWER Virtualization Best Practices RedPaper

Restoring the Virtual I/O Server from the HMC using a nim_resources.tar file

You can restore the Virtual I/O Server base code, applied fix packs, custom device drivers to support disk subsystems, and some user-defined metadata from a nim_resources.tar image stored in a remote file system.

If the system is managed by the Integrated Virtualization Manager, then you need to restore your partition profile data for the management partition and its clients before you restore the Virtual I/O Server. For instructions, see Backing up and restoring partition data. (Alternatively, you can use the rstprofdata command.)

To restore the Virtual I/O Server from a nim_resources.tar image in a file system, complete the following steps:

1. Run the installios command from the HMC command line. This restores a backup image, nim_resources.tar, that was created using the backupios command.
2. Follow the installation procedures according to the system prompts. The source of the installation images is the exported directory from the backup procedure. For example, `server1:/export/ios_backup`.
3. When the restoration is finished, open a virtual terminal connection (for example, using telnet) to the Virtual I/O Server that you restored. Some additional user input might be required.
4. If you restored the Virtual I/O Server to a different system from which it was backed up, you must restore the user-defined virtual devices. For instructions, see "Restoring user-defined virtual devices" on page 136.

Related information

IBM System p Advanced POWER Virtualization Best Practices RedPaper

Restoring the Virtual I/O Server from a NIM server using a mksysb file

You can restore the Virtual I/O Server base code, applied fix packs, custom device drivers to support disk subsystems, and some user-defined metadata from a mksysb image stored in a remote file system.

Before you start, complete the following tasks:

- Ensure that the server to which you plan to restore the Virtual I/O Server is defined as a Network Installation Management (NIM) resource.
- Ensure that the mksysb file (that contains the backup of the Virtual I/O Server) is on the NIM server.
- If the system is managed by the Integrated Virtualization Manager, then you need to restore your partition profile data for the management partition and its clients before you restore the Virtual I/O Server. For instructions, see Backing up and restoring partition data. (Alternatively, you can use the rstprofdata command.)

To restore the Virtual I/O Server from a mksysb image in a file system, complete the following tasks:

1. Define the mksysb file as a NIM resource, specifically, a NIM object, by running the nim command. To view a detailed description of the nim command, see nim Command. For example:

   ```
   nim -o define -t mksysb -a server=servername -alocation=/export/ios_backup/
   filename.mksysb objectname
   ```

   Where:

   - *servername* is the name of the server that holds the NIM resource.
   - *filename* is the name of the mksysb file.
   - *objectname* is the name by which NIM registers and recognizes the mksysb file.
2. Define a Shared Product Object Tree (SPOT) resource for the mksysb file by running the nim command. For example:

   ```
   nim -o define -t spot -a server=servername -a location=/export/ios_backup/
   SPOT -a source=objectname SPOTname
   ```

   Where:

   - *servername* is the name of the server that holds the NIM resource.
   - *objectname* is the name by which NIM registers and recognizes the mksysb file.
   - *SPOTname* is the NIM object name for the mksysb image that was created in the previous step.
3. Install the Virtual I/O Server from the mksysb file using the smit command. For example:

   ```
   smit nim_bosinst
   ```

   Ensure the following entry fields contain the following specifications.

Table 38. Specifications for the SMIT command

Field	**Specification**
Installation TYPE	mksysb
SPOT	*SPOTname* from step 3
MKSYSB	*objectname* from step 2
Remain NIM client after install?	no

4. Start the Virtual I/O Server logical partition. For instructions, see step 3, Boot the Virtual I/O Server, of Installing the Virtual I/O Server using NIM.
5. If you restored the Virtual I/O Server to a different system from which it was backed up, you must restore the user-defined virtual devices. For instructions, see “Restoring user-defined virtual devices.”

Related information

IBM System p Advanced POWER Virtualization Best Practices RedPaper

Using the NIM define operation

Defining a SPOT resource

Installing a client using NIM

Restoring user-defined virtual devices

In addition to restoring the Virtual I/O Server, you might need to restore user-defined virtual devices (such as virtual device mappings). For example, in the event of a system failure, system migration, or disaster.

User-defined virtual devices include metadata, such as virtual device mappings, that define the relationship between the physical environment and the virtual environment. In situations where you plan to restore the Virtual I/O Server to a new or different system (for example, in the event of a system failure or disaster), you need to back up both the Virtual I/O Server and user-defined virtual devices.

Before you start, restore the Virtual I/O Server from tape, DVD, or a remote file system. For instructions, see one of the following procedures:

- “Restoring the Virtual I/O Server from tape” on page 134
- “Restoring the Virtual I/O Server from one or more DVDs” on page 134
- “Restoring the Virtual I/O Server from the HMC using a nim_resources.tar file” on page 135
- “Restoring the Virtual I/O Server from a NIM server using a mksysb file” on page 135

To restore user-defined virtual devices, complete the following steps:

1. List all of the backed-up volume groups (or storage pools) by running the following command:

   ```
   restorevgstruct -ls
   ```

   This command lists the files located in the **/home/ios/vgbackups** directory.
2. Run the lspv command to determine which disks are empty.
3. Restore the volume groups (or storage pools) to the empty disks by running the following command for each volume group (or storage pool):

   ```
   restorevgstruct -vg volumegroup hdiskx
   ```

   Where:

   - *volumegroup* is the name of a volume group (or storage pool) from step 1.
   - *hdiskx* is the name of an empty disk from step 2.
4. Re-create the mappings between the virtual devices and physical devices (including storage device mappings, shared Ethernet and Ethernet adapter mappings, and virtual LAN settings) using the mkvdev command. You can find mapping information in the file that you specified in the tee command from the backup procedure. For example, `/home/padmin/`*`filename`*.

Related information

IBM System p Advanced POWER Virtualization Best Practices RedPaper

Restoring the Virtual I/O Server using IBM Tivoli Storage Manager

You can use the IBM Tivoli Storage Manager to restore the mksysb image of the Virtual I/O Server.

You can restore the Virtual I/O Server to the system from which it was backed up, or to a new or different system (for example, in the event of a system failure or disaster). The following procedure applies to restoring the Virtual I/O Server to the system from which it was backed up. First, you restore the mksysb image to the Virtual I/O Server using the dsmc command on the Tivoli Storage Manager client. But restoring the mksysb image does not restore the Virtual I/O Server. You then need to transfer the mksysb image to another system and convert the mksysb image to an installable format.

To restore the Virtual I/O Server to a new or different system, use one of the following procedures:

- “Restoring the Virtual I/O Server from tape” on page 134
- “Restoring the Virtual I/O Server from one or more DVDs” on page 134
- “Restoring the Virtual I/O Server from the HMC using a nim_resources.tar file” on page 135
- “Restoring the Virtual I/O Server from a NIM server using a mksysb file” on page 135

Before you start, complete the following tasks:

1. Ensure that the system to which you plan to transfer the mksysb image is running AIX.
2. Ensure that the system running AIX has a DVD-RW or CD-RW drive.
3. Ensure that AIX has the cdrecord and mkisofs RPMs downloaded and installed. To download and install the RPMs, see the AIX Toolbox for Linux Applications Web site.

Restriction: Interactive mode is not supported on the Virtual I/O Server. You can view session information by typing `dsmc` on the Virtual I/O Server command line.

To restore the Virtual I/O Server using Tivoli Storage Manager, complete the following tasks:

1. Determine which file you want to restore by running the dsmc command to display the files that have been backed up to the Tivoli Storage Manager server:

   ```
   dsmc -query
   ```
2. Restore the mksysb image using the dsmc command. For example:

   ```
   dsmc -restore sourcefilespec
   ```

   Where *sourcefilespec* is the directory path to the location where you want to restore the mksysb image. For example, /home/padmin/mksysb_image
3. Transfer the mksysb image to a server with a DVD-RW or CD-RW drive by running the following File Transfer Protocol (FTP) commands:
   a. Run the following command to make sure that the FTP server is started on the Virtual I/O Server: `startnetsvc ftp`
   b. Run the following command to make sure that the FTP server is started on the Virtual I/O Server: `startnetsvc ftp`
   c. Open an FTP session to the server with the DVD-RW or CD-RW drive: `ftp` *`server_hostname`*, where *server_hostname* is the hostname of the server with the DVD-RW or CD-RW drive.
   d. At the FTP prompt, change to the installation directory to the directory where you want to save the mksysb image.
   e. Set the transfer mode to binary: `binary`
   f. Turn off interactive prompting if it is on: `prompt`
   g. Transfer the mksysb image to the server: `mput mksysb_image`
   h. Close the FTP session, after transferring mksysb image, by typing `quit`.
4. Write the mksysb image to CD or DVD using the mkcd or mkdvd commands.
5. Reinstall the Virtual I/O Server using the CD or DVD that you just created. For instructions, see "Restoring the Virtual I/O Server from one or more DVDs" on page 134.

Related reference

mkcd Command

mkdvd Command

Installing or replacing a PCI adapter with the system power on in Virtual I/O Server

You can install or replace a PCI adapter in the Virtual I/O Server logical partition or in the Integrated Virtualization Manager management partition.

The Virtual I/O Server includes a PCI Hot Plug Manager that is similar to the PCI Hot Plug Manager in the AIX operating system. The PCI Hot Plug Manager allows you to hot plug PCI adapters into the server and then activate them for the logical partition without having to reboot the system. Use the PCI Hot Plug Manager for adding, identifying, or replacing PCI adapters in the system that are currently assigned to the Virtual I/O Server.

Getting started

Prerequisites:

- If you are installing a new adapter, an empty system slot must be assigned to the Virtual I/O Server logical partition. This task can be done through dynamic logical partitioning (DLPAR) operations.
  - If you are using a Hardware Management Console (HMC), you must also update the logical partition profile of the Virtual I/O Server so that the new adapter is configured to the Virtual I/O Server after you restart the system.

  - If you are using the Integrated Virtualization Manager, an empty slot is probably already assigned to the Virtual I/O Server logical partition because all slots are assigned to the Virtual I/O Server by default. You only need to assign an empty slot to the Virtual I/O Server logical partition if you previously assigned all empty slots to other logical partitions.
- If you are installing a new adapter, ensure that you have the software required to support the new adapter and determine whether there are any existing PTF prerequisites to install. To do this, use the IBM Prerequisite Web site at http://www-912.ibm.com/e_dir/eServerPrereq.nsf
- If you need help determining the PCI slot in which to place a PCI adapter, see the PCI adapter placement for machine types 82xx and 91xx or the PCI adapter placement for machine type 94xx.

Follow these steps to access the Virtual I/O Server, PCI Hot Plug Manager:
1. If you are using the Integrated Virtualization Manager, connect to the command-line interface.
2. Use the **diagmenu** command to open the Virtual I/O Server diagnostic menu. The menus are similar to the AIX diagnostic menus.
3. Select **Task Selection**, then press Enter.
4. At the Task Selection list, select **PCI Hot Plug Manager**.

Installing a PCI adapter

To install a PCI adapter with the system power on in Virtual I/O Server, do the following steps:
1. From the PCI Hot Plug Manager, select **Add a PCI Hot Plug Adapter**, then press Enter. The Add a Hot-Plug Adapter window is displayed.
2. Select the appropriate empty PCI slot from those listed, and press Enter. A fast-blinking amber LED located at the back of the server near the adapter indicates that the slot has been identified.
3. Follow the instructions on the screen to install the adapter until the LED for the specified PCI slot is set to the Action state. The adapter installation is performed the same as in a stand-alone AIX logical partition and includes the following sequence of events:
   a. Set the adapter LED to the action state so that the indicator light for the adapter slot flashes
   b. Physically install the adapter
   c. Finish the adapter installation task in **diagmenu**.
4. Run the **cfgdev** command to configure the device for the Virtual I/O Server.

If you are installing a PCI, Fibre Channel adapter, it is now ready to be attached to a SAN and have LUNs assigned to the Virtual I/O Server for virtualization.

Replacing a PCI Adapter

Prerequisite: Before you can remove or replace a storage adapter, you must unconfigure that adapter. See "Unconfiguring storage adapters" on page 140 for instructions.

To replace a PCI adapter with the system power on in Virtual I/O Server, do the following steps:
1. From the PCI Hot Plug Manager, select **Unconfigure a Device**, then press Enter.
2. Press F4 (or Esc +4) to display the **Device Names** menu.
3. Select the adapter you are removing in the **Device Names** menu.
4. In the **Keep Definition** field, use the Tab key to answer `Yes`. In the **Unconfigure Child Devices** field, use the Tab key again to answer `YES`, then press Enter.
5. Press Enter to verify the information on the **ARE YOU SURE** screen. Successful unconfiguration is indicated by the `OK` message displayed next to the Command field at the top of the screen.
6. Press F4 (or Esc +4) twice to return to the Hot Plug Manager.
7. Select **replace/remove PCI Hot Plug adapter**.
8. Select the slot that has the device to be removed from the system.

9. Select **replace**. A fast-blinking amber LED located at the back of the machine near the adapter indicates that the slot has been identified.
10. Press Enter which places the adapter in the action state, meaning it is ready to be removed from the system.

Unconfiguring storage adapters

Before you can remove or replace a storage adapter, you must unconfigure that adapter. Storage adapters are generally parent devices to media devices, such as disk drives or tape drives. Removing the parent requires that all attached child devices either be removed or placed in the define state.

Unconfiguring a storage adapter involves the following tasks:

- Closing all applications that are using the adapter you are removing, replacing, or moving
- Unmounting file systems
- Ensuring that all devices connected to the adapter are identified and stopped
- Listing all slots that are currently in use or a slot that is occupied by a specific adapter
- Identifying the adapter's slot location
- Making parent and child devices unavailable
- Making the adapter unavailable

If the adapter supports physical volumes that are in use by a client logical partition, then You can perform steps on the client logical partition before unconfiguring the storage adapter. For instructions, see "Preparing the client logical partitions." For example, the adapter might be in use because the physical volume was used to create a virtual target device, or it might be part of a volume group used to create a virtual target device.

Follow these steps to unconfigure SCSI, SSA, and Fibre Channel storage adapters:

1. Connect to the Virtual I/O Server command-line interface.
2. Use the oem_setup_env command to close all applications that are using the adapter you are unconfiguring.
3. Type `lsslot-c pci` to list all the hot plug slots in the system unit and display their characteristics.
4. Type `lsdev -C` to list the current state of all the devices in the system unit.
5. Type `unmount` to unmount previously mounted file systems, directories, or files using this adapter.
6. Type `rmdev -l adapter -R` to make the adapter unavailable.

   Attention: Do not use the -d flag with the rmdev command for hot plug operations because this action removes your configuration.

Preparing the client logical partitions

If the virtual target devices of the client logical partitions are not available, the client logical partitions can fail or they might be unable to perform I/O operations for a particular application. If you use the HMC to manage the system, you might have redundant Virtual I/O Server logical partitions, which allow for Virtual I/O Server maintenance and avoid downtime for client logical partitions. If you are replacing an adapter on the Virtual I/O Server and your client logical partition is dependent on one or more of the physical volumes accessed by that adapter, then You can take action on the client before you unconfigure the adapter.

The virtual target devices must be in the define state before the Virtual I/O Server adapter can be replaced. Do not remove the virtual devices permanently.

To prepare the client logical partitions so that you can unconfigure an adapter, complete the following steps depending on your situation.

Table 39. Situations and steps for preparing the client logical partitions

Situation	Steps
You have redundant hardware on the Virtual I/O Server for the adapter.	No action is required on the client logical partition.
HMC-managed systems only: You have redundant Virtual I/O Server logical partitions that, in conjunction with virtual client adapters, provide multiple paths to the physical volume on the client logical partition.	No action is required on the client logical partition. However, path errors might be logged on the client logical partition.
HMC-managed systems only: You have redundant Virtual I/O Server logical partitions that, in conjunction with virtual client adapters, provide multiple physical volumes that are used to mirror a volume group.	See the procedures for your client operating system. For example, for AIX, see Replacing a disk on the Virtual I/O Server in the IBM System p Advanced POWER Virtualization Best Practices Redpaper. The procedure for Linux is similar to this procedure for AIX.
You do not have redundant Virtual I/O Server logical partitions.	Shut down the client logical partition. For instructions, see the following topics about shutting down logical partitions: • For systems that are managed by the HMC, see "Shutting down AIX logical partitions using the HMC", "Shutting down IBM i logical partitions using the HMC", and "Shutting down Linux logical partitions using the HMC" in the Logical partitioning.[1] • For systems that are managed by the Integrated Virtualization Manager, see "Shutting down logical partitions."
[1]The Logical partitioning can be found on the Hardware Information Web site at http://publib.boulder.ibm.com/infocenter/systems/scope/hw/topic/iphdx/power_systems.htm .	

Shutting down logical partitions

You can use the Integrated Virtualization Manager to shut down the logical partitions or to shut down the entire managed system.

Use any role other than View Only to perform this task.

The Integrated Virtualization Manager provides the following types of shutdown options for logical partitions:

- Operating System (recommended)
- Delayed
- Immediate

The recommended shutdown method is to use the client operating systems shutdown command. Use the immediate shutdown method only as a last resort because using this method causes an abnormal shutdown which might result in data loss.

If you choose the Delayed shutdown method, then be aware of the following considerations:

- Shutting down the logical partitions is equivalent to pressing and holding the white control-panel power button on a server that is not partitioned.
- Use this procedure only if you cannot successfully shut down the logical partitions through operating system commands. When you use this procedure to shut down the selected logical partitions, the logical partitions wait a predetermined amount of time to shut down. This allows the logical partitions time to end jobs and write data to disks. If the logical partition is unable to shut down within the predetermined amount of time, it ends abnormally, and the next restart might take a long time.

If you plan to shut down the entire managed system, shut down each client logical partition, then shut down the Virtual I/O Server management partition.

To shut down a logical partition, complete the following steps in the Integrated Virtualization Manager:

1. In the navigation area, select **View/Modify Partitions** under **Partition Management**. The View/Modify Partitions page is displayed.
2. Select the logical partition that you want to shut down.
3. From the Tasks menu, click **Shutdown**. The Shutdown Partitions page is displayed.
4. Select the shutdown type.
5. Optional: Select **Restart after shutdown completes** if you want the logical partition to start immediately after it shuts down.
6. Click **OK** to shut down the partition. The View/Modify Partitions page is displayed, and the logical partition state has a value of shut down.

Viewing information and statistics about the Virtual I/O Server, the server, and virtual resources

You can view information and statistics about the Virtual I/O Server, the server, and virtual resources to help you manage and monitor the system, and troubleshoot problems.

The following table lists the information and statistics available on the Virtual I/O Server, as well as the commands you need to run to view the information and statistics.

Table 40. Information and associated commands for the Virtual I/O Server

Information to view	**Command**
Statistics about kernel threads, virtual memory, disks, traps, and processor activity.	vmstat
Statistics for a Fibre Channel device driver.	fcstat
A summary of virtual memory usage.	svmon
Information about the Virtual I/O Server and the server, such as the server model, machine ID, Virtual I/O Server logical partition name and ID, and the LAN network number.	uname

Table 40. Information and associated commands for the Virtual I/O Server (continued)

Information to view	Command
Generic and device-specific statistics for an Ethernet driver or device, including the following information for a Shared Ethernet Adapter: • Shared Ethernet Adapter statistics: – Number of real and virtual adapters (If you are using Shared Ethernet Adapter failover, this number does not include the control channel adapter) – Shared Ethernet Adapter flags – VLAN IDs – Information about real and virtual adapters • Shared Ethernet Adapter failover statistics: – High availability statistics – Packet types – State of the Shared Ethernet Adapter – Bridging mode • GARP VLAN Registration Protocol (GVRP) statistics: – Bridge Protocol Data Unit (BPDU) statistics – Generic Attribute Registration Protocol (GARP) statistics – GARP VLAN Registration Protocol (GVRP) statistics • Listing of the individual adapter statistics for the adapters associated with the Shared Ethernet Adapter	enstat

The vmstat, fcstat, svmon, and uname commands are available with Virtual I/O Server version 1.5 or later. To update the Virtual I/O Server, see “Updating the Virtual I/O Server” on page 126.

Monitoring the Virtual I/O Server

You can monitor the Virtual I/O Server using error logs or IBM Tivoli Monitoring.

Error logs

AIX and Linux client logical partitions log errors against failing I/O operations. Hardware errors on the client logical partitions associated with virtual devices usually have corresponding errors logged on the server. However, if the failure is within the client logical partition, there will not be errors on the server. Also, on Linux client logical partitions, if the algorithm for retrying SCSI temporary errors is different from the algorithm used by AIX, the errors might not be recorded on the server.

IBM Tivoli Monitoring

With Virtual I/O Server V1.3.0.1 (fix pack 8.1), you can install and configure the IBM Tivoli Monitoring System Edition for System p agent on the Virtual I/O Server. With Tivoli Monitoring System Edition for System p, you can monitor the health and availability of multiple System p servers (including the Virtual I/O Server) from the Tivoli Enterprise Portal. Tivoli Monitoring System Edition for System p gathers data from the Virtual I/O Server, including data about physical volumes, logical volumes, storage pools, storage mappings, network mappings, real memory, processor resources, mounted file system sizes, and so on. From the Tivoli Enterprise Portal, you can view a graphical representation of the data, use predefined thresholds to alert you on key metrics, and resolve issues based on recommendations provided by the Expert Advice feature of Tivoli Monitoring.

Security on the Virtual I/O Server

Become familiar with the Virtual I/O Server security features.

Beginning with version 1.3 of the Virtual I/O Server, you can set security options that provide tighter security controls over your Virtual I/O Server environment. These options allow you to select a level of system security hardening and specify the settings allowable within that level. The Virtual I/O Server security feature also allows you to control network traffic by enabling the Virtual I/O Server firewall. You can configure these options using the viosecure command. To help you set up system security when you initially install the Virtual I/O Server, the Virtual I/O Server provides the configuration assistance menu. You can access the configuration assistance menu by running the cfgassist command.

Using the viosecure command, you can set, change, and view current security settings. By default, no Virtual I/O Server security levels are set. You must run the viosecure command to change the settings.

The following sections provide an overview of these features.

Virtual I/O Server system security hardening

The system security hardening feature protects all elements of a system by tightening security or implementing a higher level of security. Although hundreds of security configurations are possible with the Virtual I/O Server security settings, you can easily implement security controls by specifying a high, medium, or low security level.

Using the system security hardening features provided by Virtual I/O Server, you can specify values such as the following:

- Password policy settings
- usrck, pwdck, grpck, and sysck actions
- Default file-creation settings
- Settings included in the crontab command

Configuring a system at too high a security level might deny services that are needed. For example, telnet and rlogin are disabled for high level security because the login password is sent over the network unencrypted. If a system is configured at too low a security level, the system might be vulnerable to security threats. Since each enterprise has its own unique set of security requirements, the predefined High, Medium, and Low security configuration settings are best suited as a starting point for security configuration rather than an exact match for the security requirements of a particular enterprise. As you become more familiar with the security settings, you can make adjustments by choosing the hardening rules you want to apply. You can get information about the hardening rules by running the man command.

Virtual I/O Server firewall

Using the Virtual I/O Server firewall, you can enforce limitations on IP activity in your virtual environment. With this feature, you can specify which ports and network services are allowed access to the Virtual I/O Server system. For example, if you need to restrict login activity from an unauthorized port, you can specify the port name or number and specify deny to remove it from the allow list. You can also restrict a specific IP address.

Connecting to the Virtual I/O Server using OpenSSH

You can set up remote connections to the Virtual I/O Server using secure connections.

You can use the Open Source Secure Sockets Layer (OpenSSL) and Portable Secure Shell (OpenSSH) software to connect to the Virtual I/O Server using secure connections. For more information about OpenSSL and OpenSSH, see the OpenSSL Project and Portable SSH Web sites.

To connect to the Virtual I/O Server using OpenSSH, complete the following tasks:

1. If you are using a version of Virtual I/O Server prior to version 1.3.0, then install OpenSSH before you connect. For instructions, see "Downloading, installing, and updating OpenSSH and OpenSSL" on page 146.
2. Connect to the Virtual I/O Server. If you are using version 1.3.0 or later, then connect using either an interactive or noninteractive shell. If you are using a version prior to 1.3.0, then connect using only an interactive shell.
   - To connect using an interactive shell, type the following command from the command line of a remote system:

     ```
     ssh username@vioshostname
     ```

     where *username* is your user name for the Virtual I/O Server and *vioshostname* is the name of the Virtual I/O Server.
   - To connect using a noninteractive shell, run the following command:

     ```
     ssh username@vioshostname command
     ```

     Where:
     - *username* is your user name for the Virtual I/O Server.
     - *vioshostname* is the name of the Virtual I/O Server.
     - *command* is the command that you want to run. For example, `ioscli lsmap -all`.

       Note: When using a noninteractive shell, remember to use the full command form (including the `ioscli` prefix) for all Virtual I/O Server commands.
3. Authenticate SSH. If you are using version 1.3.0 or later, then authenticate using either passwords or keys. If you are using a version prior to 1.3.0, then authenticate using only passwords.
   - To authenticate using passwords, enter your user name and password when prompted by the SSH client.
   - To authenticate using keys, perform the following steps on the SSH client's operating system:
     a. Create a directory called $HOME/.ssh to store the keys. You can use RSA or DSA keys.
     b. Run the **ssh-keygen** command to generate public and private keys. For example,

        ```
        ssh-keygen -t  rsa
        ```

        This creates the following files in the $HOME/.ssh directory:
        - Private key: id_rsa
        - Public key: id_rsa.pub

     c. Run the following command to append the public key to the `authorized_keys2` file on the Virtual I/O Server:

        ```
        cat $HOME/.ssh/public_key_file | ssh username@vioshostname tee -a /home/username/.ssh/authorized_keys2
        ```

        Where:
        - *public_key_file* is the public key file that is generated in the previous step. For example, id_rsa.pub.
        - *username* is your user name for the Virtual I/O Server.
        - *vioshostname* is the name of the Virtual I/O Server.

The Virtual I/O Server might not include the latest version of OpenSSH or OpenSSL with each release. In addition, there might be OpenSSH or OpenSSL updates released in between Virtual I/O Server releases.

In these situations, you can update OpenSSH and OpenSSL on the Virtual I/O Server by downloading and installing OpenSSH and OpenSSL. For instructions, see "Downloading, installing, and updating OpenSSH and OpenSSL."

Downloading, installing, and updating OpenSSH and OpenSSL

If you are using a Virtual I/O Server version prior to 1.3, you must download and install OpenSSH and OpenSSL software before you can connect to the Virtual I/O Server using OpenSSH. You can also use this procedure to update OpenSSH and OpenSSL on the Virtual I/O Server.

OpenSSH and OpenSSL might need to be updated on your Virtual I/O Server if the Virtual I/O Server did not include the latest version of OpenSSH or OpenSSL, or if there were OpenSSH or OpenSSL updates released in between Virtual I/O Server releases. In these situations, you can update OpenSSH and OpenSSL on the Virtual I/O Server by downloading and installing OpenSSH and OpenSSL using the following procedure.

For more information about OpenSSL and OpenSSH, see the OpenSSL Project and Portable SSH Web sites.

Downloading the Open Source software:

The OpenSSL software contains the encrypted library that is required to use the OpenSSH software. To download the software, complete the following tasks:

1. Download the OpenSSL RPM package to your workstation or host computer.
   a. To get the RPM package, go to the AIX Toolbox for Linux Applications Web site and click the **AIX Toolbox Cryptographic Content** link on the right side of the Web page.
   b. If you are registered to download the RPM packages, then sign in and accept the license agreement.
   c. If you are not registered to download the RPM packages, then complete the registration process and accept the license agreement. After registering, you are redirected to the download page.
   d. Select any version of the package for download: **openssl - Secure Sockets Layer and cryptography libraries and tools** and click **Download Now** to start the download.
2. Download the OpenSSH software by completing the following steps:

   Note: Alternatively, you can install the software from the AIX Expansion Pack.
   a. From your workstation (or host computer), go to the SourceFORGE.net Web site.
   b. Click **Download OpenSSH on AIX** to view the latest file releases.
   c. Select the appropriate download package and click **Download**.
   d. Click the openssh package (tar.Z file) to continue with the download.
3. Create a directory on the Virtual I/O Server for the Open Source software files. For example, to create an installation directory named install_ssh, run the following command: `mkdir install_ssh`.
4. Transfer the software packages to the Virtual I/O Server by running the following File Transfer Protocol (FTP) commands from the computer on which you downloaded the software packages:
   a. Run the following command to make sure that the FTP server is started on the Virtual I/O Server: `startnetsvc ftp`
   b. Open an FTP session to the Virtual I/O Server on your local host: `ftp` *`vios_server_hostname`*, where *vios_server_hostname* is the hostname of the Virtual I/O Server.
   c. At the FTP prompt, change to the installation directory to the directory that you created for the Open Source files: `cd` *`install_ssh`*, where *install_ssh* is the directory that contains the Open Source files.
   d. Set the transfer mode to binary: `binary`
   e. Turn off interactive prompting if it is on: `prompt`

f. Transfer the downloaded software to the Virtual I/O Server: `mput` *`ssl_software_pkg`*, where *ssl_software_pkg* is the software that you downloaded.
g. Close the FTP session, after transferring both software packages, by typing `quit`.

Install the Open Source software on the Virtual I/O Server:

To install the software, complete the following steps:

1. Run the following command from the Virtual I/O Server command line: `updateios -dev` *`install_ssh`* `-accept -install`, where *install_ssh* is the directory that contains the Open Source files. The installation program automatically starts the Secure Shell daemon (sshd) on the server.
2. Begin using the **ssh** and **scp** commands; no further configuration is required.

   Restrictions:
   - The **sftp** command is not supported on versions of Virtual I/O Server earlier than 1.3.
   - Noninteractive shells are not supported using OpenSSH with the Virtual I/O Server versions earlier than 1.3.

Configuring Virtual I/O Server system security hardening

Set the security level to specify security hardening rules for your Virtual I/O Server system.

To implement system security hardening rules, you can use the viosecure command to specify a security level of high, medium, or low. A default set of rules is defined for each level. You can also set a level of default, which returns the system to the system standard settings and removes any level settings that have been applied.

The low level security settings are a subset of the medium level security settings, which are a subset of the high level security settings. Therefore, the *high* level is the most restrictive and provides the greatest level of control. You can apply all of the rules for a specified level or select which rules to activate for your environment. By default, no Virtual I/O Server security levels are set; you must run the viosecure command to modify the settings.

Use the following tasks to configure the system security settings.

Setting a security level

To set a Virtual I/O Server security level of high, medium, or low, use the command `viosecure -level`. For example:

```
viosecure -level low -apply
```

Changing the settings in a security level

To set a Virtual I/O Server security level in which you specify which hardening rules to apply for the setting, run the viosecure command interactively. For example:

1. At the Virtual I/O Server command line, type `viosecure -level high`. All the security level options (hardening rules) at that level are displayed ten at a time (pressing Enter displays the next set in the sequence).
2. Review the options displayed and make your selection by entering the numbers, separated by a comma, that you want to apply, or type **ALL** to apply all the options or **NONE** to apply none of the options.
3. Press **Enter** to display the next set of options, and continue entering your selections.

   Note: To exit the command without making any changes, type “q”.

Viewing the current security setting

To display the current Virtual I/O Server security level setting use the viosecure command with the -view flag. For example:

```
viosecure -view
```

Removing security level settings

- To unset any previously set system security levels and return the system to the standard system settings, run the following command: `viosecure -level default`
- To remove the security settings that have been applied, run the following command: `viosecure -undo`

Configuring Virtual I/O Server firewall settings

Enable the Virtual I/O Server firewall to control IP activity.

The Virtual I/O Server firewall is not enabled by default. To enable the Virtual I/O Server firewall, you must turn it on by using the viosecure command with the -firewall option. When you enable it, the default setting is activated, which allows access for the following IP services:

- ftp
- ftp-data
- ssh
- web
- https
- rmc
- cimom

Note: The firewall settings are contained in the file viosecure.ctl in the /home/ios/security directory. If for some reason the viosecure.ctl file does not exist when you run the command to enable the firewall, you receive an error. You can use the -force option to enable the standard firewall default ports.

You can use the default setting or configure the firewall settings to meet the needs of your environment by specifying which ports or port services to allow. You can also turn off the firewall to deactivate the settings.

Use the following tasks at the Virtual I/O Server command line to configure the Virtual I/O Server firewall settings:

1. Enable the Virtual I/O Server firewall by running the following command:
   ```
   viosecure -firewall on
   ```
2. Specify the ports to allow or deny, by using the following command:
   ```
   viosecure -firwall allow | deny -port number
   ```
3. View the current firewall settings by running the following command:
   ```
   viosecure -firewall view
   ```
4. If you want to disable the firewall configuration, run the following command:
   ```
   viosecure -firewall off
   ```

Configuring a Kerberos client on the Virtual I/O Server

You can configure a Kerberos client on the Virtual I/O Server to enhance security in communications across the Internet.

Before you start, ensure that the Virtual I/O Server version 1.5 or later. To update the Virtual I/O Server, see "Updating the Virtual I/O Server" on page 126.

Kerberos is a network authentication protocol that provides authentication for client and server applications by using a secret-key cyrptography. It negotiates authenticated, and optionally encrypted, communications between two points anywhere on the Internet. Kerberos authentication generally works as follows:

1. A Kerberos client sends a request for a ticket to the Key Distribution Center (KDC).
2. The KDC creates a ticket-granting ticket (TGT) for the client and encrypts it using the client's password as the key.
3. The KDC returns the encrypted TGT to the client.
4. The client attempts to decrypt the TGT, using its password.
5. If the client successfully decrypts the TGT (for example, if the client gives the correct password), the client keeps the decrypted TGT. The TGT indicates proof of the client's identity.

To configure a Kerberos client on the Virtual I/O Server, run the follwoing command.

```
mkkrb5clnt -c KDC_server -r realm_name \ -s Kerberos_server -d Kerberos_client
```

Where:

- *KDC_server* is the name of the KDC server.
- *realm_name* is the name of the realm to which you want to configure the Kerberos client.
- *Kerberos_server* is the fully qualified host name of the Kerberos server.
- *Kerberos_client* is the domain name of the Kerberos client.

For example:

```
mkkrb5clnt -c bob.kerberso.com -r KERBER.COM \ -s bob.kerberso.com -d testbox.com
```

In this example, you configure the Kerberos client, testbox.com, to the Kerberos server, bob.kerberso.com. The KDC is running on bob.kerberso.com.

Managing users on the Virtual I/O Server

You can create, list, change, switch, and remove users by using Virtual I/O Server or the IBM Tivoli Identity Manager.

When the Virtual I/O Server is installed, the only user type that is active is the prime administrator (**padmin**). The prime administrator can create additional user IDs with types of system administrator, service representative, or development engineer.

Note: You cannot create the prime administrator (**padmin**) user ID. It is automatically created and enabled after the Virtual I/O Server is installed.

The following table lists the user management tasks available on the Virtual I/O Server, as well as the commands you must run to accomplish each task.

Table 41. Tasks and associated commands for working with Virtual I/O Server users

Task	Command
Change passwords	cfgassist
Create a system administrator user ID	mkuser
Create a service representative (SR) user ID	mkuser with the **-sr** flag
Create a development engineer (DE) user ID	mkuser with the **-de** flag
Create an LDAP user	mkuser with the **-ldap** flag
List a user's attributes For example, determine whether a user is an LDAP user.	lsuser

Table 41. Tasks and associated commands for working with Virtual I/O Server users (continued)

Task	Command
Change a user's attributes	chuser
Switch to another user	su
Remove a user	rmuser

You can use the IBM Tivoli Identity Manager to automate the management of Virtual I/O Server users. Tivoli Identity Manager provides a Virtual I/O Server adapter that acts as an interface between the Virtual I/O Server and the Tivoli Identity Manager Server. The adapter acts as a trusted virtual administrator on the Virtual I/O Server, performing tasks like the following:

- Creating a user ID to authorize access to the Virtual I/O Server.
- Modifying an existing user ID to access the Virtual I/O Server.
- Removing access from a user ID. This deletes the user ID from the Virtual I/O Server.
- Suspending a user account by temporarily deactivating access to the Virtual I/O Server.
- Restoring a user account by reactivating access to the Virtual I/O Server.
- Changing a user account password on the Virtual I/O Server.
- Reconciling the user information of all current users on the Virtual I/O Server.
- Reconciling the user information of a particular user account on the Virtual I/O Server by performing a lookup.

For more information, see the IBM Tivoli Identity Manager product manuals.

Troubleshooting the Virtual I/O Server

Find information about diagnosing Virtual I/O Server problems and information about how to correct those problems.

This section includes information about troubleshooting the Virtual I/O Server. For information about troubleshooting the Integrated Virtualization Manager, see Troubleshooting the Integrated Virtualization Manager.

Troubleshooting the Virtual I/O Server logical partition

Find information and procedures for troubleshooting and diagnosing the Virtual I/O Server logical partition.

Troubleshooting virtual SCSI problems

Find information and procedures for troubleshooting virtual SCSI problems in the Virtual I/O Server.

For problem determination and maintenance, use the diagmenu command provided by the Virtual I/O Server.

If you are still having problems after using the diagmenu command, contact your next level of support and ask for assistance.

Refer to the AIX fast-path problem-isolation documentation in the Service provider information because, in certain cases, the diagnostic procedures described in the AIX fast-path problem-isolation documentation are not available from the diagmenu command menu.

Correcting a failed Shared Ethernet Adapter configuration

You can troubleshoot errors that occur when you configure a Shared Ethernet Adapter, such as those that result in message 0514-040, by using the lsdev, netstat, and entstat commands.

When you configure a Shared Ethernet Adapter the configuration can fail with the following error:

```
Method error (/usr/lib/methods/cfgsea):
        0514-040 Error initializing a device into the kernel.
```

To correct the problem, complete the following steps:

1. Verify that the physical and virtual adapters that are being used to create the shared Ethernet adapter are available by running the following command:

   ```
   lsdev -type adapter
   ```
2. Make sure that the interface of neither the physical nor any of the virtual adapters are configured. Run the following command:

   ```
   netstat -state
   ```

   Important: None of the interfaces of the adapters must be listed in the output. If any interface name (for example, *en0*) does is listed in the output, detach it as follows:

   ```
   chdev -dev interface_name -attr state=detach
   ```

   You might want to perform this step from a console connection because it is possible that detaching this interface will end your network connection to the Virtual I/O Server.
3. Verify that the virtual adapters that are used for data are trunk adapters by running the following command:

   ```
   entstat -all entX | grep Trunk
   ```

   Note:
   - The trunk adapter does not apply to the virtual adapter that is used as the control channel in a Shared Ethernet Adapter Failover configuration.
   - If any of the virtual adapters that are used for data are not trunk adapters, you need to enable them to access external networks from the HMC.
4. Verify that the physical device and the virtual adapters in the Shared Ethernet Adapter are in agreement on the checksum offload setting:
   a. Determine the checksum offload setting on physical device by running the following command:

      ```
      lsdev -dev device_name -attr chksum_offload
      ```

      where *device_name* is the name of the physical device. For example, ent0.
   b. If `chksum_offload` is set to `yes`, enable checksum offload for all of the virtual adapters in the Shared Ethernet Adapter by running the following command:

      ```
      chdev -dev device_name -attr chksum_offload=yes
      ```

      Where *device_name* is the name of a virtual adapter in the Shared Ethernet Adapter. For example, ent2.
   c. If `chksum_offload` is set to `no`, disable checksum offload for all of the virtual adapters in the Shared Ethernet Adapter by running the following command:

      ```
      chdev -dev device_name -attr chksum_offload=no
      ```

      where *device_name* is the name of a virtual adapter in the Shared Ethernet Adapter.
   d. If there is no output, the physical device does not support checksum offload and therefore does not have the attribute. To resolve the error, disable checksum offload for all of the virtual adapters in the Shared Ethernet Adapter by running the following command:

      ```
      chdev -dev device_name -attr chksum_offload=no
      ```

where *device_name* is the name of a virtual adapter in the Shared Ethernet Adapter.

5. If the real adapter is a Host Ethernet Adapter port, also known as, a Logical Integrated Virtual Ethernet adapter port, make sure that the Virtual I/O Server has been configured as the promiscuous logical partition for the physical port of the logical Integrated Virtual Ethernet adapter from the HMC.

Debugging problems with Ethernet connectivity

You can determine Ethernet connectivity problems by examining Ethernet statistics produced by the entstat command. Then, you can debug the problems using the starttrace and stoptrace commands.

To help debug problems with Ethernet connectivity, follow these steps:

1. Verify that the source client logical partition can ping another client logical partition on the same system without going through the Virtual I/O Server. If this fails, the problem is likely in the client logical partition's virtual Ethernet setup. If the ping is successful, proceed to the next step.
2. Start a ping on the source logical partition to a destination machine so that the packets are sent through the Virtual I/O Server. This ping will most likely fail. Proceed to the next step with the ping test running.
3. On the Virtual I/O Server, type the following command:

   ```
   entstat –all SEA_adapter
   ```

   where *SEA_adapter* is the name of your Shared Ethernet Adapter.
4. Verify that the VLAN ID to which the logical partition belongs is associated with the correct virtual adapter in the VLAN IDs section of the output. Examine the `ETHERNET STATISTICS` for the virtual adapter for this VLAN and verify that the packet counts under the `Receive statistics` column are increasing.

   This verifies that the packets are being received by the Virtual I/O Server through the correct adapter. If the packets are not being received, the problem might be in the virtual adapter configuration. Verify the VLAN ID information for the adapters using the Hardware Management Console (HMC).
5. Examine the `ETHERNET STATISTICS` for the physical adapter for this VLAN and verify that the packet counts under the `Transmit statistics` column are increasing. This step verifies that the packets are being sent out of the Virtual I/O Server.
   - If this count is increasing, then the packets are going out of the physical adapter. Continue to step 6.
   - If this count is not increasing, then the packets are not going out of the physical adapter, and to further debug the problem, you must begin the system trace utility. Follow the instructions in step 9 to collect a system trace, statistical information, and the configuration description. Contact service and support if you need to debug the problem further.
6. Verify that the target system outside (on physical side of Virtual I/O Server) is receiving packets and sending out replies. If this is not happening, either the wrong physical adapter is associated with the Shared Ethernet Adapter or the Ethernet switch might not be configured correctly.
7. Examine the `ETHERNET STATISTICS` for the physical adapter for this VLAN and verify that the packet counts under the Receive statistics column are increasing. This step verifies that the ping replies are being received by the Virtual I/O Server. If this count is not increasing, the switch might not be configured correctly.
8. Examine the `ETHERNET STATISTICS` for the virtual adapter for this VLAN and verify that the packet counts under the `Transmit statistics` column are increasing. This step verifies that the packet is being transmitted by the Virtual I/O Server through the correct virtual adapter. If this count is not increasing, start the system trace utility. Follow the instructions in step 9 to collect a system trace, statistical information, and the configuration description. Work with service and support to debug the problem further.
9. Use the Virtual I/O Server trace utility to debug connectivity problems. Start a system trace using the starttrace command specifying the trace hook ID. The trace hook ID for Shared Ethernet Adapter is 48F. Use the stoptrace command to stop the trace. Use the cattracerpt command to read the trace log, format the trace entries, and write a report to standard output.

Enabling noninteractive shells on Virtual I/O Server 1.3 or later

After upgrading the Virtual I/O Server to 1.3 or later, you can enable noninteractive shells using the startnetsvc command.

If you installed OpenSSH on a level of the Virtual I/O Server prior to 1.3, and then upgraded to 1.3 or later, noninteractive shells might not work because the SSH configuration file needs modification.

To enable noninteractive shells in Virtual I/O Server 1.3 or later, run the following command from the SSH client:

```
ioscli startnetsvc ssh
```

Note: You can run the startnetsvc command when the SSH service is running. In this situation, the command appears to fail, but is successful.

Recovering when disks cannot be located

Learn how to recover from disks not displaying when trying to boot or install a client logical partition.

Occasionally, the disk that is needed to install the client logical partition cannot be located. In this situation, if the client is already installed, start the client logical partition. Ensure that you have the latest levels of the software and firmware. Then ensure that the **Slot number** of the virtual SCSI server adapter matches the **Remote partition virtual slot number** of the virtual SCSI client adapter.

1. Ensure that you have the latest levels of the Hardware Management Console, firmware, and Virtual I/O Server. Follow these steps:
   a. To check whether you have the latest level of the HMC, see the Installing and configuring the Hardware Management Console. To view the PDF file of Installing and configuring the Hardware Management Console, approximately 3 MB in size, see

      http://publib.boulder.ibm.com/infocenter/systems/scope/hw/topic/iphai/iphai.pdf .
   b. Ensure that you have the latest firmware.
   c. To check whether you have the latest level of the Virtual I/O Server, see "Updating the Virtual I/O Server" on page 126.
2. Ensure the server virtual SCSI adapter slot number is mapped correctly to the client logical partition remote slot number:
   a. In the navigation area, expand **Systems Management** → **Servers** and click the server on which the Virtual I/O Server logical partition is located.
   b. In the contents area, select the Virtual I/O Server logical partition.
   c. Click **Tasks** and select **Properties**.
   d. Click the **Virtual Adapters** tab.
   e. Click **Virtual SCSI**.
   f. If the values of the **Remote Partition** and **Remote Adapter** are **Any Partition** and **Any Partition Slot**, then complete the following steps:
      - Expand **Virtual SCSI** and click the slot number.
      - Select **Only selected client partition can connect**.
      - Enter the client logical partition's ID and adapter and click **OK**
      - Click **Virtual SCSI**.
   g. Record values of the **Remote Partition** and **Remote Adapter**. These values represent the client logical partition and the slot number of the client's virtual SCSI adapter that can connect to the associated server adapter. For example, the values of **Remote Partition**, **Remote Adapter**, and **Adapter** are as follows: AIX_client, 2, 3. This means that virtual SCSI adapter 2 on the client logical partition AIX_client can connect to the Virtual I/O Server virtual SCSI adapter 3.
   h. Repeat steps a through g for the client logical partition.

3. Ensure the server virtual SCSI adapter slot number is mapped correctly to the client logical partition remote slot number. Follow these steps:
   a. Right-click the server profile, and select **Properties**.
   b. Click the Virtual I/O Server tab.
   c. If the **Only selected remote partition and slot can connect** radio button is not selected, select it.
   d. Note the **Remote partition** and **Remote partition virtual slot number** values. This shows the client logical partition name and the client logical partition virtual slot number. This is the client logical partition and slot number that can connect to the slot given in the **Slot number** dialog box at the top of the **Virtual SCSI Adapter Properties** window.
   e. Repeat items a through e in this step for the client logical partition.
4. The **Adapter** value on the client logical partition must match the **Remote Adapter** on the Virtual I/O Server logical partition, and the **Adapter** value on the Virtual I/O Server logical partition must match the **Remote Adapter** on the client logical partition. If these numbers do not match, from the HMC, modify the profile properties to reflect the correct mapping.
5. From the Virtual I/O Server command line, type `cfgdev`.
6. Shut down and reactivate the client logical partition.
7. From the Virtual I/O Server command line, type `lsmap -all`. You see results similar to the following:

   ```
   SVSA            Physloc                                      Client Partition ID
   --------------- -------------------------------------------- ------------------
   vhost0          U9113.550.10BE8DD-V1-C3                      0x00000002

   VTD                   vhdisk0
   LUN                   0x8100000000000000
   Backing device        hdisk5
   Physloc               U787B.001.DNW025F-P1-C5-T1-W5005076300C10899-L536F000000000000
   ```

   In this example, the client logical partition ID is 2 (`0x00000002`).

   Note: If the client logical partition is not yet installed, the Client Partition ID is 0x00000000.
   The slot number of the server SCSI adapter is displayed under Physloc column. The digits following the `-C` specify the slot number. In this case, the slot number is 3.
8. From the Virtual I/O Server command line, type `lsdev -virtual`. You see results similar to the following:

   ```
   name            status     description

   vhost0          Available  Virtual SCSI Server Adapter

   vhdisk0         Available  Virtual Target Device - Disk
   ```

Troubleshooting AIX client logical partitions

Find information and procedures for troubleshooting AIX client logical partitions.

If your client partition is using virtual I/O resources, check the Service Focal Point and Virtual I/O Server first to ensure that the problem is not on the server.

On client partitions running the current level of AIX, when a hardware error is logged on the server and a corresponding error is logged on the client partition, the Virtual I/O Server provides a correlation error message in the error report.

Run the following command to gather an error report:

```
errpt -a
```

Running the **errpt** command returns results similar to the following:

```
LABEL:          VSCSI_ERR2
IDENTIFIER:     857033C6

Date/Time:       Tue Feb 15 09:18:11 2005
Sequence Number: 50
Machine Id:      00C25EEE4C00
Node Id:         vio_client53A
Class:           S
Type:            TEMP
Resource Name:   vscsi2

Description
Underlying transport error

Probable Causes
PROCESSOR

Failure Causes
PROCESSOR

        Recommended Actions
        PERFORM PROBLEM DETERMINATION PROCEDURES
Detail Data
Error Log Type
01
Reserve
00
Error Number
0006
RC
0000 0002
VSCSI Pointer
```

Compare the LABEL, IDENTIFIER, and Error Number values from your error report to the values in the following table to help identify the problem and determine a resolution.

Table 42. Labels, identifiers, error numbers, problem descriptions, and resolutions of common virtual SCSI client logical partition problems

Label	Identifier	Error Number	Problem	Resolution
VSCSI_ERR2	857033C6	0006 RC 0000 0002	The virtual SCSI server adapter on the Virtual I/O Server logical partition is not open.	Make the server adapter on the Virtual I/O Server logical partition available for use.
		001C RC 0000 0000	The virtual SCSI server adapter on the Virtual I/O Server logical partition has been closed abruptly.	Determine why the server adapter in the Virtual I/O Server logical partition was closed.

Table 42. Labels, identifiers, error numbers, problem descriptions, and resolutions of common virtual SCSI client logical partition problems (continued)

Label	Identifier	Error Number	Problem	Resolution
VSCSI_ERR3	ED995F18	000D RC FFFF FFF0	The virtual SCSI server adapter on the Virtual I/O Server logical partition is being used by another client logical partition.	Terminate the client logical partition that is using the server adapter.
		000D RC FFFF FFF9	The virtual SCSI server adapter (partition number and slot number) specified in the client adapter definition does not exist.	On the HMC, correct the client adapter definition to associate it with a valid server adapter.

Reference information for the Virtual I/O Server

Find reference information about the Virtual I/O Server commands, configuration attributes for Tivoli agents and clients, networking statistics and attributes, and Virtual I/O Server user types.

Virtual I/O Server and Integrated Virtualization Manager command descriptions

You can view a description of each Virtual I/O Server and Integrated Virtualization Manager command.

See the Virtual I/O Server and Integrated Virtualization Manager commands. To view the PDF file of Virtual I/O Server and Integrated Virtualization Manager commands, approximately 4 MB in size, see http://publib.boulder.ibm.com/infocenter/systems/scope/hw/topic/iphcg/iphcg.pdf .

Configuration attributes for IBM Tivoli agents and clients

Learn about required and optional configuration attributes and variables for the IBM Tivoli Monitoring agent, the IBM Tivoli Usage and Accounting Manager agent, the IBM Tivoli Storage Manager client, and the IBM TotalStorage Productivity Center agents.

In the following tables, the term *attribute* refers to an option that you can add to a Virtual I/O Server command. The term *variable* refers to an option that you can specify in a configuration file for Tivoli Storage Manager or Tivoli Usage and Accounting Manager.

IBM Tivoli Monitoring

Table 43. Tivoli Monitoring configuration attributes

Attribute	Description
HOSTNAME	The host name or IP address of the Tivoli Enterprise Monitoring Server (TEMS) server to which the monitoring agent sends data.

Table 43. Tivoli Monitoring configuration attributes (continued)

Attribute	Description
MANAGING_SYSTEM	The host name or IP address of the Hardware Management Console (HMC) attached to the managed system on which the Virtual I/O Server with the monitoring agent is located. You can specify only one HMC per monitoring agent. If you do not specify the MANAGING_SYSTEM attribute, the Virtual I/O Server uses the Resource Monitoring and Control (RMC) connection to obtain the host name of IP address of the HMC. If the monitoring agent is running on the Integrated Virtualization Manager, then you do not need to specify the MANAGING_SYSTEM attribute.
RESTART_ON_REBOOT	Determines whether the monitoring agent restarts whenever the Virtual I/O Server restarts. `TRUE` indicates that the monitoring agent restarts whenever the Virtual I/O Server restarts. `FALSE` indicates that the monitoring agent does not restart whenever the Virtual I/O Server restarts.

IBM Tivoli Storage Manager

Table 44. Tivoli Storage Manager configuration attributes

Attribute	Description
SERVERNAME	The host name of the Tivoli Storage Manager server to which the Tivoli Storage Manager client is associated.
SERVERIP	The IP address or domain name of the Tivoli Storage Manager server to which the Tivoli Storage Manager client is associated.
NODENAME	The name of the machine on which the Tivoli Storage Manager client is installed.

IBM Tivoli Usage and Accounting Manager

Table 45. Tivoli Usage and Accounting Manager configuration variables in the A_config.par file

Variable	Description	Possible values	Default value
AACCT_TRANS_IDS	Designates the AIX advanced accounting record types included within the usage reports.	1, 4, 6, 7, 8, 10, 11, or 16	10
AACCT_ONLY	Determines whether the Usage and Accounting Manager agent collects accounting data.	• Y: Indicates that the Usage and Accounting Manager agent collects accounting data. • N: Indicates that the Usage and Accounting Manager agent does not collect accounting data.	Y

Table 45. Tivoli Usage and Accounting Manager configuration variables in the A_config.par file (continued)

Variable	Description	Possible values	Default value
ITUAM_SAMPLE	Determines whether the Usage and Accounting Manager agent collects data about the storage file system.	• Y: Indicates that the Usage and Accounting Manager agent collects data about the storage file system. • N: Indicates that the Usage and Accounting Manager agent does not collect data about the storage file system.	N

Table 46. Tivoli Usage and Accounting Manager configuration attributes

Attribute	Description
ACCT_DATA0	The size, in MB, of the first data file that holds daily accounting information.
ACCT_DATA1	The size, in MB, of the second data file that holds daily accounting information.
ISYSTEM	The time, in minutes, when the agent generates system interval records.
IPROCESS	The time, in minutes, when the system generates aggregate process records.

IBM TotalStorage Productivity Center attributes

Table 47. IBM TotalStorage Productivity Center configuration attributes

Attribute	Description	Required or optional
S	Host name or IP address of the TotalStorage Productivity Center Server associated with the TotalStorage Productivity Center agent.	Required
A	Host name or IP address of the Agent Manager.	Required
devAuth	Password for authentication to the TotalStorage Productivity Center device server.	Required
caPass	Password for authentication to the command agent.	Required
caPort	Number that identifies the port for the common agent. The default is 9510.	Optional
amRegPort	Number that identifies the registration port for the Agent Manager. The default is 9511.	Optional
amPubPort	Number that identifies the public port for the Agent Manager. The default is 9513.	Optional

Table 47. IBM TotalStorage Productivity Center configuration attributes (continued)

Attribute	Description	Required or optional
dataPort	Number that identifies the port for the TotalStorage Productivity Center Data server. The default is 9549.	Optional
devPort	Number that identifies the port of the TotalStorage Productivity Center Device server. The default is 9550.	Optional
newCA	The default is true.	Optional
oldCA	The default is false.	Optional
daScan	Runs a scan for the TPC_data agent after installation. The default is true.	Optional
daScript	Runs the script for the TPC_data agent after installation. The default is true.	Optional
daIntsall	Installs the TPC_data agent. The default is true.	Optional
faInstall	Installs the TPC_fabric agent. The default is true.	Optional
U	Uninstalls the TotalStorage Productivity Center agents. Possible values include: • all • data • fabric	Optional

Related information

IBM Tivoli Application Dependency Discovery Manager Information Center

IBM Tivoli Identity Manager

IBM Tivoli Monitoring version 6.2.1 documentation

IBM Tivoli Monitoring Virtual I/O Server Premium Agent User's Guide

IBM Tivoli Storage Manager

IBM Tivoli Usage and Accounting Manager Information Center

IBM TotalStorage Productivity Center Information Center

GARP VLAN Registration Protocol statistics

Learn about Bridge Protocol Data Unit (BPDU), Generic Attribute Registration Protocol (GARP), and GARP VLAN Registration Protocol (GVRP) displayed by running the entstat -all command. You can also view examples.

BPDU refers to all protocol packets that are exchanged between the switch and the Shared Ethernet Adapter. The only bridge protocol currently available with the Shared Ethernet Adapter is GARP. GARP is a generic protocol used to exchange attribute information between two entities. The only type of GARP currently available on the Shared Ethernet Adapter is GVRP. With GVRP, the attributes exchanged are VLAN values.

BPDU statistics

The BPDU statistics include all BPDU packets sent or received.

Table 48. Descriptions of BPDU statistics

BPDU statistic	Description
Transmit	**Packets** Number of packets sent. **Failed packets** Number of packets that could not be sent (for example, packets that could not be sent because there was no memory to allocate the outgoing packet).
Receive	**Packets** Number of packets received. **Unprocessed Packets** Packets that could not be processed because the protocol was not running at the time. **Non-contiguous Packets** Packets that were received in several packet fragments. **Packets with unknown PID** Packets that had a protocol ID (PID) different than GARP. A high number is typical because the switch might be exchanging other BPDU protocol packets that the Shared Ethernet Adapter does not support. **Packets with Wrong Length** Packets whose specified length (in the Ethernet header) does not match the length of the Ethernet packet received.

GARP statistics

The GARP statistics include those BPDU packets sent or received that are of type GARP.

Table 49. Descriptions of GARP statistics

GARP statistic	Description
Transmit	**Packets** Number of packets sent. **Failed packets** Number of packets that could not be sent (for example, packets that could not be sent because there was no memory to allocate the outgoing packet). **Leave All Events** Packets sent with event type *Leave All*. **Join Empty Events** Packets sent with event type *Join Empty* **Join In Events** Packets sent with event type *Join In* **Leave Empty Events** Packets sent with event type *Leave Empty* **Leave In Events** Packets sent with event type *Leave In* **Empty Events** Packets sent with event type *Empty*
Receive	**Packets** Number of packets received **Unprocessed Packets** Packets that could not be processed because the protocol was not running at the time. **Packets with Unknown Attr Type:** Packets with an unsupported attribute type. A high number is typical because the switch might be exchanging other GARP protocol packets that the Shared Ethernet Adapter does not support. For example, GARP Multicast Registration Protocol (GMRP). **Leave All Events** Packets received with event type *Leave All* **Join Empty Events** Packets received with event type *Join Empty* **Join In Events** Packets received with event type *Join In* **Leave Empty Events** Packets received with event type *Leave Empty* **Leave In Events** Packets received with event type *Leave In* **Empty Events** Packets received with event type *Empty*

GVRP statistics

The GVRP statistics include those GARP packets sent or received that are exchanging VLAN information using GVRP.

Table 50. Descriptions of GVRP statistics

GVRP statistic	Description
Transmit	**Packets** Number of packets sent **Failed packets** Number of packets that could not be sent (for example, packets that could not be sent because there was no memory to allocate the outgoing packet). **Leave All Events** Packets sent with event type *Leave All*. **Join Empty Events** Packets sent with event type *Join Empty* **Join In Events** Packets sent with event type *Join In* **Leave Empty Events** Packets sent with event type *Leave Empty* **Leave In Events** Packets sent with event type *Leave In* **Empty Events** Packets sent with event type *Empty*

Table 50. Descriptions of GVRP statistics (continued)

GVRP statistic	Description
Receive	**Packets** Number of packets received. **Unprocessed Packets** Packets that could not be processed because the protocol was not running at the time. **Packets with Invalid Length** Packets that contains one or more attributes whose length does not correspond to its event type. **Packets with Invalid Event** Packets that contain one or more attributes whose event type is invalid. **Packets with Invalid Value** Packets that contain one or more attributes whose value is invalid (for example, an invalid VLAN ID). **Total Invalid Attributes** Sum of all of the attributes that had an invalid parameter. **Total Valid Attributes** Sum of all of the attributes that had no invalid parameters. **Leave All Events** Packets sent with event type *Leave All*. **Join Empty Events** Packets sent with event type *Join Empty* **Join In Events** Packets sent with event type *Join In* **Leave Empty Events** Packets sent with event type *Leave Empty* **Leave In Events** Packets sent with event type *Leave In* **Empty Events** Packets sent with event type *Empty*

Example statistics

Running the entstat -all command returns results similar to the following:

```
--------------------------------------------------------------
Statistics for adapters in the Shared Ethernet Adapter ent3
--------------------------------------------------------------
Number of adapters: 2
SEA Flags: 00000009
    < THREAD >
    < GVRP >
VLAN IDs :
    ent2: 1
Real Side Statistics:
    Packets received: 0
    Packets bridged: 0
    Packets consumed: 0
    Packets transmitted: 0
    Packets dropped: 0
Virtual Side Statistics:
    Packets received: 0
    Packets bridged: 0
    Packets consumed: 0
    Packets transmitted: 0
    Packets dropped: 0
Other Statistics:
    Output packets generated: 0
    Output packets dropped: 0
    Device output failures: 0
    Memory allocation failures: 0
    ICMP error packets sent: 0
    Non IP packets larger than MTU: 0
    Thread queue overflow packets: 0

--------------------------------------------------------------
Bridge Protocol Data Units (BPDU) Statistics:

Transmit Statistics:                          Receive Statistics:
--------------------                          -------------------
Packets: 2                                    Packets: 1370
Failed packets: 0                             Unprocessed Packets: 0
                                              Non-contiguous Packets: 0
                                              Packets w/ Unknown PID: 1370
                                              Packets w/ Wrong Length: 0

---------------------------------------------------------------
General Attribute Registration Protocol (GARP) Statistics:

Transmit Statistic:                           Receive Statistics:
-------------------                           -------------------
Packets: 2                                    Packets: 0
Failed packets: 0                             Unprocessed Packets: 0
                                              Packets w/ Unknow Attr. Type: 0

Leave All Events: 0                           Leave All Events: 0
Join Empty Events: 0                          Join Empty Events: 0
Join In Events: 2                             Join In Events: 0
Leave Empty Events: 0                         Leave Empty Events: 0
Leave In Events: 0                            Leave In Events: 0
Empty Events: 0                               Empty Events: 0

---------------------------------------------------------------
GARP VLAN Registration Protocol (GVRP) Statistics:

Transmit Statistics:                          Receive Statistics:
--------------------                          -------------------
Packets: 2                                    Packets: 0
Failed packets: 0                             Unprocessed Packets: 0
                                              Attributes w/ Invalid Length: 0
                                              Attributes w/ Invalid Event: 0
                                              Attributes w/ Invalid Value: 0
                                              Total Invalid Attributes: 0
                                              Total Valid Attributes: 0

Leave All Events: 0                           Leave All Events: 0
Join Empty Events: 0                          Join Empty Events: 0
Join In Events: 2                             Join In Events: 0
Leave Empty Events: 0                         Leave Empty Events: 0
Leave In Events: 0                            Leave In Events: 0
Empty Events: 0                               Empty Events: 0
```

Network attributes

Find instructions for managing network attributes.

You can use several of the Virtual I/O Server commands, including chdev, mkvdev, and cfglnagg, to change device or network attributes. This section defines attributes that can be modified.

Ethernet attributes

You can modify the following Ethernet attributes.

Attribute	Description
Maximum Transmission Unit (*mtu*)	Specifies maximum transmission unit (MTU). This value can be any number from 60 through 65535, but it is media dependent.
Interface State (*state*)	**detach** Removes an interface from the network interface list. If the last interface is detached, the network interface driver code is unloaded. To change the interface route of an attached interface, that interface must be detached and added again with the **chdev -dev** *Interface* **-attr** *state=detach* command. **down** Marks an interface as inactive, which keeps the system from trying to transmit messages through that interface. Routes that use the interface, however, are not automatically disabled. (**chdev -dev** *Interface* **-attr** *state=down*) **up** Marks an interface as active. This parameter is used automatically when setting the first address for an interface. It can also be used to enable an interface after the **chdev -dev** *Interface* **-attr** *state=up* command.
Network Mask (*netmask*)	Specifies how much of the address to reserve for subdividing networks into subnetworks. The *mask* includes both the network part of the local address and the subnet part, which is taken from the host field of the address. The mask can be specified as a single hexadecimal number beginning with 0x, in standard Internet dotted-decimal notation. In the 32-bit address, the mask contains bits with a value of 1 for the bit positions reserved for the network and subnet parts, and a bit with the value of 0 for the bit positions that specify the host. The mask contains the standard network portion, and the subnet segment is contiguous with the network segment.

Shared Ethernet Adapter attributes

You can modify the following Shared Ethernet Adapter attributes.

Attribute	Description
PVID (*pvid*)	Specifies the PVID to use for the Shared Ethernet Adapter.
PVID adapter (*pvid_adapter*)	Specifies the default virtual adapter to use for non-VLAN tagged packets.
Physical adapter (*real_adapter*)	Specifies the physical adapter associated with the Shared Ethernet Adapter.

Attribute	Description
Thread (*thread*)	Activates or deactivates threading on the Shared Ethernet Adapter. Activating this option adds approximately 16 - 20% more machine cycles per transaction for MTU 1500 streaming, and approximately 31 – 38% more machine cycles per transaction for MTU 9000. The threading option adds more machine cycles per transaction at lower workloads due to the threads being started for each packet. At higher workload rates, such as full duplex or the request/response workloads, the threads can run longer without waiting and being redispatched. Threaded mode should be used when virtual SCSI will be run on the same Virtual I/O Server logical partition as Shared Ethernet Adapter. Threaded mode helps ensure that virtual SCSI and the Shared Ethernet Adapter can share the processor resource appropriately. However, threading adds more instruction path length, which uses additional processor cycles. If the Virtual I/O Server logical partition will be dedicated to running shared Ethernet devices (and associated virtual Ethernet devices) only, the adapters should be configured with threading disabled. You can enable or disable threading using the **-attr thread** option of the mkvdev command. To enable threading, use the `-attr thread=1` option. To disable threading, use the `-attr thread=0` option. For example, the following command disables threading for Shared Ethernet Adapter `ent1`: `mkvdev -sea ent1 -vadapter ent5 -default ent5 -defaultid 1 -attr thread=0`
Virtual adapters (*virt_adapter*)	Lists the virtual Ethernet adapters associated with the Shared Ethernet Adapter.
TCP segmentation offload (*largesend*)	Enables TCP largesend capability (also known as segmentation offload) from logical partitions to the physical adapter. The physical adapter must be enabled for TCP largesend for the segmentation offload from the logical partition to the Shared Ethernet Adapter to work. Also, the logical partition must be capable of performing a largesend operation. On AIX, largesend can be enabled on a logical partition using the ifconfig command. You can enable or disable TCP largesend using the -a largesend option of the chdev command. To enable it, use the ’-a largesend=1’ option. To disable it, use the ’-a largesend=0’ option. For example, the following command enables *largesend* for Shared Ethernet Adapter ent1: `chdev -l ent1 -a largesend=1` By default the setting is disabled (largesend=0).
Jumbo frames (*jumbo_frames*)	Allows the interface configured over the Shared Ethernet Adapter to increase its MTU to 9000 bytes (the default is 1500). If the underlying physical adapter does not support jumbo frames and the *jumbo_frames* attribute is set to `yes`, then configuration fails. The underlying physical adapter must support jumbo frames. The Shared Ethernet Adapter automatically enables jumbo frames on its underlying physical adapter if *jumbo_frames* is set to `yes`. You cannot change the value of *jumbo_frames* at run time.
GARP VLAN Registration Protocol (GVRP) (*gvrp*)	Enables and disables GVRP on a Shared Ethernet Adapter.

Shared Ethernet Adapter failover attributes

You can modify the following Shared Ethernet Adapter failover attributes.

Attribute	Description
High availability mode (*ha_mode*)	Determines whether the devices participate in a failover setup. The default is `disabled`. Typically, a Shared Ethernet Adapter in a failover setup is operating in `auto` mode, and the primary adapter is decided based on which adapter has the highest priority (lowest numerical value). A shared Ethernet device can be forced into the standby mode, where it will behave as the backup device as long as it can detect the presence of a functional primary.
Control Channel (*ctl_chan*)	Sets the virtual Ethernet device that is required for a Shared Ethernet Adapter in a failover setup so that it can communicate with the other adapter. There is no default value for this attribute, and it is required when the *ha_mode* is not set to `disabled`.
Internet address to ping (*netaddr*)	Optional attribute that can be specified for a Shared Ethernet Adapter that has been configured in a failover setup. When this attribute is specified, a shared Ethernet device will periodically ping the IP address to verify connectivity (in addition to checking for link status of the physical devices). If it detects a loss of connectivity to the specified ping host, it will initiate a failover to the backup Shared Ethernet Adapter. This attribute is not supported when you use a Shared Ethernet Adapter with a Host Ethernet Adapter (or Integrated Virtual Ethernet).

INET attributes

You can modify the following INET attributes.

Attribute	Description
Host Name (*hostname*)	Specify the host name that you want to assign to the current machine. When specifying the host name, use ASCII characters, preferably alphanumeric only. Do not use a period in the host name. Avoid using hexadecimal or decimal values as the first character (for example `3Comm`, where `3C` might be interpreted as a hexadecimal character). For compatibility with earlier hosts, use an unqualified host name of fewer than 32 characters. If the host uses a domain name server for name resolution, the host name must contain the full domain name. In the hierarchical domain naming system, names consist of a sequence of subnames that are not case-sensitive and that are separated by periods with no embedded blanks. The DOMAIN protocol specifies that a local domain name must be fewer than 64 characters, and that a host name must be fewer than 32 characters in length. The host name is given first. Optionally, the full domain name can be specified; the host name is followed by a period, a series of local domain names separated by periods, and finally by the root domain. A fully specified domain name for a host, including periods, must be fewer than 255 characters in length and in the following form: `host.subdomain.subdomain.rootdomain` In a hierarchical network, certain hosts are designated as name servers that resolve names into Internet addresses for other hosts. This arrangement has two advantages over the flat name space: resources of each host on the network are not consumed in resolving names, and the person who manages the system does not need to maintain name-resolution files on each machine on the network. The set of names managed by a single name server is known as its *zone of authority*.
Gateway (*gateway*)	Identifies the gateway to which packets are addressed. The *Gateway* parameter can be specified either by symbolic name or numeric address.

Attribute	Description
Route (*route*)	Specifies the route. The format of the *Route* attribute is: *route*=*destination*, *gateway*, [*metric*]. **destination** Identifies the host or network to which you are directing the route. The *Destination* parameter can be specified either by symbolic name or numeric address. **gateway** Identifies the gateway to which packets are addressed. The *Gateway* parameter can be specified either by symbolic name or numeric address. **metric** Sets the routing metric. The default is 0 (zero). The routing metric is used by the routing protocol (the *routed* daemon). Higher metrics have the effect of making a route less favorable. Metrics are counted as additional hops to the destination network or host.

Adapter attributes

You can modify the following adapter attributes. The attribute behavior can vary, based on the adapter and driver you have.

Attribute	Adapters/Drivers	Description
Media Speed (*media_speed*)	• 2-Port 10/100/1000 Base-TX PCI-X Adapter • 10/100/1000 Base-T Ethernet PCI-X Adapter Device Driver	The media speed attribute indicates the speed at which the adapter attempts to operate. The available speeds are 10 Mbps half-duplex, 10 Mbps full-duplex, 100 Mbps half-duplex, 100 Mbps full-duplex and autonegotiation, with a default of autonegotiation. Select auto-negotiate when the adapter should use autonegotiation across the network to determine the speed. When the network will not support autonegotiation, select the specific speed. 1000 MBps half and full duplex are not valid values. According to the IEEE 802.3z specification, gigabit speeds of any duplexity must be autonegotiated for copper (TX)-based adapters. If these speeds are desired, select auto-negotiate.
Media Speed (*media_speed*)	• 2-Port Gigabit Ethernet-SX PCI-X Adapter • Gigabit Ethernet-SX PCI-X Adapter Device Driver	The media speed attribute indicates the speed at which the adapter attempts to operate. The available speeds are 1000 Mbps full-duplex and autonegotiation. The default is autonegotiation. Select auto-negotiate when the adapter should use autonegotiation across the network to determine the duplexity. When the network does not support autonegotiation, select 1000 Mbps full-duplex.

Attribute	Adapters/Drivers	Description
Media Speed (*media_speed*)	- 10/100 Mbps Ethernet PCI Adapter Device Driver	The media speed attribute indicates the speed at which the adapter attempts to operate. The available speeds are 10 Mbps half-duplex, 10 Mbps full-duplex, 100 Mbps half-duplex, 100 Mbps full-duplex and autonegotiation, with a default of autonegotiation. When the adapter should use autonegotiation across the network to determine the speed, select autonegotiate. When the network will not support autonegotiation, select the specific speed. If autonegotiation is selected, the remote link device must also be set to autonegotiate to ensure the link works correctly.
Media Speed (*media_speed*)	- 10/100/1000 Base-T Ethernet PCI adapter - Gigabit Ethernet-SX PCI Adapter Device Driver	The media speed attribute indicates the speed at which the adapter attempts to operate. The available speeds are 10 Mbps half-duplex, 10 Mbps full-duplex, 100 Mbps half-duplex, 100 Mbps full-duplex and autonegotiation, with a default of autonegotiation. Select autonegotiate when the adapter should use autonegotiation across the network to determine the speed. When the network will not support autonegotiation, select the specific speed. For the adapter to run at 1000 Mbit/s, the autonegotiation setting must be selected. **Note:** For the Gigabit Ethernet-SX PCI Adapter, the only selection available is autonegotiation.
Enable Alternate Ethernet Address (*use_alt_addr*)		Setting this attribute to `yes` indicates that the address of the adapter, as it appears on the network, is the one specified by the Alternate Ethernet Address attribute. If you specify the `no` value, the unique adapter address written in a ROM on the adapter card is used. The default value is `no`.
Alternate Ethernet Address (*alt_addr*)		Allows the adapter unique address, as it appears on the LAN network, to be changed. The value entered must be an Ethernet address of 12 hexadecimal digits and must not be the same as the address of any other Ethernet adapter. There is no default value. This field has no effect unless the Enable Alternate Ethernet Address attribute is set to `yes` value, in which case this field must be filled in. A typical Ethernet address is 0x02608C000001. All 12 hexadecimal digits, including leading zeros, must be entered.
Enable Link Polling (*poll_link*)	- 10/100Mbps Ethernet PCI Adapter Device Driver	Select `no` to cause the device driver to poll the adapter to determine the status of the link at a specified time interval. The time interval value is specified in the **Poll Link Time Interval** field. If you select `no`, the device driver will not poll the adapter for its link status. The default value is `no`.

Attribute	Adapters/Drivers	Description
Poll Link Time Interval (*poll_link_time*)	• 10/100Mbps Ethernet PCI Adapter Device Driver	The amount of time, in milliseconds, between polls to the adapter for its link status that the device driver is allowed. This value is required when the **Enable Link Polling** option is set to `yes`. A value between 100 through 1000 can be specified. The incremental value is 10. The default value is 500.
Flow Control (*flow_ctrl*)	• 10/100/1000 Base-T Ethernet PCI-X Adapter Device Driver • Gigabit Ethernet-SX PCI-X Adapter Device Driver • 2-Port 10/100/1000 Base-TX PCI-X Adapter • 2-Port Gigabit Ethernet-SX PCI-X Adapter • Gigabit Ethernet-SX PCI Adapter Device Driver	This attribute specifies whether the adapter should enable transmit and receive flow control. The default value is `no`.
Transmit Jumbo Frames (*jumbo_frames*)	• 10/100/1000 Base-T Ethernet PCI-X Adapter Device Driver • Gigabit Ethernet-SX PCI-X Adapter Device Driver • 2-Port 10/100/1000 Base-TX PCI-X Adapter • 2-Port Gigabit Ethernet-SX PCI-X Adapter • Gigabit Ethernet-SX PCI Adapter Device Driver	Setting this attribute to `yes` indicates that frames up to 9018 bytes in length might be transmitted on this adapter. If you specify `no`, the maximum size of frames transmitted is 1518 bytes. Frames up to 9018 bytes in length can always be received on this adapter.

Attribute	Adapters/Drivers	Description
Checksum Offload (*chksum_offload*)	• 10/100/1000 Base-T Ethernet PCI-X Adapter Device Driver • Gigabit Ethernet-SX PCI-X Adapter Device Driver • 2-Port 10/100/1000 Base-TX PCI-X Adapter • 2-Port Gigabit Ethernet-SX PCI-X Adapter • Gigabit Ethernet-SX PCI Adapter Device Driver • Virtual Ethernet adapters	Setting this attribute to `yes` indicates that the adapter calculates the checksum for transmitted and received TCP frames. If you specify `no`, the checksum will be calculated by the appropriate software. When a virtual Ethernet adapter has checksum offload enabled, the adapter advertises it to the hypervisor. The hypervisor tracks which virtual Ethernet adapters have checksum offload enabled and manages inter-partition communication accordingly. When network packets are routed through the Shared Ethernet Adapter, there is a potential for link errors. In this environment, the packets must traverse the physical link with a checksum. Communication works in the following way: • When a packet is received from the physical link, the physical adapter verifies the checksum. If the packet's destination is a virtual Ethernet adapter with checksum offload enabled, the receiver does not have to perform checksum verification. A receiver that does not have checksum offload enabled will accept the packet after checksum verification. • When a packet originates from a virtual Ethernet adapter with checksum offload enabled, it travels to the physical adapter without a checksum. The physical adapter will generate a checksum before sending the packet out. Packets originating from a virtual Ethernet adapter with checksum offload disabled generate the checksum at the source. To enable checksum offload for a Shared Ethernet Adapter, all constituent devices must have it enabled as well. The shared Ethernet device will fail if the underlying devices do not have the same checksum offload settings.
Enable Hardware Transmit TCP Resegmentation (*large_send*)	• 10/100/1000 Base-T Ethernet PCI-X Adapter Device Driver • Gigabit Ethernet-SX PCI-X Adapter Device Driver • 2-Port 10/100/1000 Base-TX PCI-X Adapter • 2-Port Gigabit Ethernet-SX PCI-X Adapter • Gigabit Ethernet-SX PCI Adapter Device Driver	This attribute specifies whether the adapter is to perform transmit TCP resegmentation for TCP segments. The default value is `no`.

Link Aggregation (EtherChannel) device attributes

You can modify the following Link Aggregation, or EtherChannel, attributes.

Attribute	Description
Link Aggregation adapters (*adapter_names*)	The adapters that currently make up the Link Aggregation device. If you want to modify these adapters, modify this attribute and select all the adapters that should belong to the Link Aggregation device. When you use this attribute to select all of the adapters that should belong to the Link Aggregation device, its interface must not have an IP address configured.
Mode (*mode*)	The type of channel that is configured. In standard mode, the channel sends the packets to the adapter based on an algorithm (the value used for this calculation is determined by the Hash Mode attribute). In round_robin mode, the channel gives one packet to each adapter before repeating the loop. The default mode is standard. Using the 802.3ad mode, the Link Aggregation Control Protocol (LACP) negotiates the adapters in the Link Aggregation device with an LACP-enabled switch. If the Hash Mode attribute is set to anything other than the default, this attribute must be set to standard or 802.3ad. Otherwise, the configuration of the Link Aggregation device will fail.
Hash Mode (*hash_mode*)	If operating under standard or IEEE 802.3ad mode, the hash mode attribute determines how the outgoing adapter for each packet is chosen. Following are the different modes: • `default`: uses the destination IP address to determine the outgoing adapter. • `src_port`: uses the source TCP or UDP port for that connection. • `dst_port`: uses the destination TCP or UDP port for that connection. • `src_dst_port`: uses both the source and destination TCP or UDP ports for that connection to determine the outgoing adapter. You cannot use round-robin mode with any hash mode value other than default. The Link Aggregation device configuration will fail if you attempt this combination. If the packet is not TCP or UDP, it uses the default hashing mode (destination IP address). Using TCP or UDP ports for hashing can make better use of the adapters in the Link Aggregationdevice, because connections to the same destination IP address can be sent over different adapters (while still retaining the order of the packets), thus increasing the bandwidth of the Link Aggregation device.
Internet Address to Ping (*netaddr*)	This field is optional. The IP address that the Link Aggregation device should ping to verify that the network is up. This is only valid when there is a backup adapter and when there are one or more adapters in the Link Aggregation device. An address of zero (or all zeros) is ignored and disables the sending of ping packets if a valid address was previously defined. The default is to leave this field blank.
Retry Timeout (*retry_time*)	This field is optional. It controls how often the Link Aggregation device sends out a ping packet to poll the current adapter for link status. This is valid only when the Link Aggregation device has one or more adapters, a backup adapter is defined, and the **Internet Address to Ping** field contains a non-zero address. Specify the timeout value in seconds. The range of valid values is 1 to 100 seconds. The default value is 1 second.
Number of Retries (*num_retries*)	This field is optional. It specifies the number of lost ping packets before the Link Aggregation device switches adapters. This is valid only when the Link Aggregation device has one or more adapters, a backup adapter is defined, and the **Internet Address to Ping** field contains a non-zero address. The range of valid values is 2 to 100 retries. The default value is 3.

Attribute	Description
Enable Gigabit Ethernet Jumbo Frames (*use_jumbo_frame*)	This field is optional. To use this attribute, all of the underlying adapters, as well as the switch, must support jumbo frames. This will work only with a Standard Ethernet (en) interface, not an IEEE 802.3 (et) interface.
Enable Alternate Address (*use_alt_addr*)	This field is optional. If you set this to yes, you can specify a MAC address that you want the Link Aggregation device to use. If you set this option to no, the Link Aggregation device uses the MAC address of the first adapter.
Alternate Address (*alt_addr*)	If **Enable Alternate Address** is set to `yes`, specify the MAC address that you want to use. The address you specify must start with `0x` and be a 12-digit hexadecimal address.

VLAN attributes

You can modify the following VLAN attributes.

Attribute	Value
VLAN Tag ID (*vlan_tag_id*)	The unique ID associated with the VLAN driver. You can specify from 1 to 4094.
Base Adapter (*base_adapter*)	The network adapter to which the VLAN device driver is connected.

Shared Ethernet Adapter Quality of Service (QoS) attribute

You can modify the following qos_mode attribute.

disabled mode

This is the default mode. VLAN traffic is not inspected for the priority field. For example,

```
chdev -dev <sea device name> -attr qos_mode=disabled
```

strict mode

More important traffic is bridged over less important traffic. This mode provides better performance and more bandwidth to more important traffic; however, it can result in substantial delays for less important traffic. For example,

```
chdev -dev <sea device name> -attr qos_mode=strict
```

loose mode

A cap is placed on each priority level, so that after a number of bytes are sent for each priority level, the next level is serviced. This method ensures that all packets will eventually be sent. More important traffic is given less bandwidth with this mode than with strict mode; however, the caps in loose mode are such that more bytes are sent for the more important traffic, so it still gets more bandwidth than less important traffic. For example,

```
chdev -dev <sea device name> -attr qos_mode=loose
```

Client-specific Shared Ethernet Adapter statistics

To gather network statistics at a client level, enable advanced accounting on the Shared Ethernet Adapter to provide more information about its network traffic. To enable client statistics, set the Shared Ethernet Adapter accounting attribute to enabled (the default value is disabled). When advanced accounting is enabled, the Shared Ethernet Adapter keeps track of the hardware (MAC) addresses of all of the packets it receives from the LPAR clients, and increments packet and byte counts for each client independently. After advanced accounting is enabled on the Shared Ethernet Adapter, you can generate a report to view per-client statistics by running the seastat command.

Note: Advanced accounting must be enabled on the Shared Ethernet Adapter before you can use the seastat command to print any statistics.
To enable advanced accounting on the Shared Ethernet Adapter, enter the following command:

```
chdev -dev <sea device name> -attr accounting=enabled
```

The following command displays per-client Shared Ethernet Adapter statistics. The optional -n flag disables name resolution on IP addresses.

```
seastat -d <sea device name> [-n]
```

The following command clears all of the per-client Shared Ethernet Adapter statistics that have been gathered:

```
seastat -d <sea device name> -c
```

Shared Ethernet Adapter failover statistics

Learn about Shared Ethernet Adapter failover statistics, such as high availability information and packet types, and view examples.

Statistic descriptions

Table 51. Descriptions of Shared Ethernet Adapter failover statistics

Statistic	Description
High availability	**Control Channel PVID** Port VLAN ID of the virtual Ethernet adapter used as the control channel. **Control Packets in** Number of packets received on the control channel. **Control Packets out** Number of packets sent on the control channel.
Packet types	**Keep-Alive Packets** Number of keep-alive packets received on the control channel. Keep-alive packets are received on the backup Shared Ethernet Adapter while the primary Shared Ethernet Adapter is active. **Recovery Packets** Number of recovery packets received on the control channel. Recovery packets are sent by the primary Shared Ethernet Adapter when it recovers from a failure and is ready to be active again. **Notify Packets** Number of notify packets received on the control channel. Notify packets are sent by the backup Shared Ethernet Adapter when it detects that the primary Shared Ethernet Adapter has recovered. **Limbo Packets** Number of limbo packets received on the control channel. Limbo packets are sent by the primary Shared Ethernet Adapter when it detects that its physical network is not operational, or when it cannot ping the specified remote host (to inform the backup that it needs to become active).

Table 51. Descriptions of Shared Ethernet Adapter failover statistics (continued)

Statistic	Description
State	The current state of the Shared Ethernet Adapter. **INIT** The Shared Ethernet Adapter failover protocol has just been initiated. **PRIMARY** The Shared Ethernet Adapter is actively connecting traffic between the VLANs to the network. **BACKUP** The Shared Ethernet Adapter is idle and not connecting traffic between the VLANs and the network. **RECOVERY** The primary Shared Ethernet Adapter recovered from a failure and is ready to be active again. **NOTIFY** The backup Shared Ethernet Adapter detected that the primary Shared Ethernet Adapter recovered from a failure and that it needs to become idle again. **LIMBO** One of the following situations is true: • The physical network is not operational. • The physical network's state is unknown. • The Shared Ethernet Adapter cannot ping the specified remote host.
Bridge Mode	Describes to what level, if any, the Shared Ethernet Adapter is currently bridging traffic. **Unicast** The Shared Ethernet Adapter is only sending and receiving unicast traffic (no multicast or broadcast traffic). To avoid broadcast storms, the Shared Ethernet Adapter sends and receives unicast traffic only while it is in the INIT or the RECOVERY states. **All** The Shared Ethernet Adapter is sending and receiving all types of network traffic. **None** The Shared Ethernet Adapter is not sending or receiving any network traffic.
Number of Times Server became Backup	Number of times the Shared Ethernet Adapter was active and became idle because of a failure.
Number of Times Server became Primary	Number of times the Shared Ethernet Adapter was idle and became active because the primary Shared Ethernet Adapter failed.

Table 51. Descriptions of Shared Ethernet Adapter failover statistics (continued)

Statistic	Description
High Availability Mode	How the Shared Ethernet Adapter behaves regarding the Shared Ethernet Adapter failover protocol. **Auto** The Shared Ethernet Adapter failover protocol determines whether the Shared Ethernet Adapter acts as the primary Shared Ethernet Adapter or as the backup Shared Ethernet Adapter. **Standby** The Shared Ethernet Adapter operates as a backup if there is another Shared Ethernet Adapter available to act as the primary. *Standby* causes a primary Shared Ethernet Adapter to become a backup Shared Ethernet Adapter if there is another Shared Ethernet Adapter that can become the primary Shared Ethernet Adapter. **Priority** Specifies the trunk priority of the virtual Ethernet adapters of the Shared Ethernet Adapter. It is used by the Shared Ethernet Adapter protocol to determine which Shared Ethernet Adapter acts as the primary Shared Ethernet Adapter and which Shared Ethernet Adapter acts as the backup Shared Ethernet Adapter. Values range from 1 to 12, where a lower number is favored to act as a primary Shared Ethernet Adapter.

Example statistics

Running the entstat -all command returns results similar to the following:

```
ETHERNET STATISTICS (ent8) :
Device Type: Shared Ethernet Adapter
Hardware Address: 00:0d:60:0c:05:00
Elapsed Time: 3 days 20 hours 34 minutes 26 seconds

Transmit Statistics:                          Receive Statistics:
--------------------                          -------------------
Packets: 7978002                              Packets: 5701362
Bytes: 919151749                              Bytes: 664049607
Interrupts: 3                                 Interrupts: 5523380
Transmit Errors: 0                            Receive Errors: 0
Packets Dropped: 0                            Packets Dropped: 0
                                              Bad Packets: 0
Max Packets on S/W Transmit Queue: 2
S/W Transmit Queue Overflow: 0
Current S/W+H/W Transmit Queue Length: 1

Elapsed Time: 0 days 0 hours 0 minutes 0 seconds
Broadcast Packets: 5312086                    Broadcast Packets: 3740225
Multicast Packets: 265589                     Multicast Packets: 194986
No Carrier Sense: 0                           CRC Errors: 0
DMA Underrun: 0                               DMA Overrun: 0
Lost CTS Errors: 0                            Alignment Errors: 0
Max Collision Errors: 0                       No Resource Errors: 0
Late Collision Errors: 0                      Receive Collision Errors: 0
Deferred: 0                                   Packet Too Short Errors: 0
```

```
SQE Test: 0                                    Packet Too Long Errors: 0
Timeout Errors: 0                              Packets Discarded by Adapter: 0
Single Collision Count: 0                      Receiver Start Count: 0
Multiple Collision Count: 0
Current HW Transmit Queue Length: 1

General Statistics:
-------------------
No mbuf Errors: 0
Adapter Reset Count: 0
Adapter Data Rate: 0
Driver Flags: Up Broadcast Running
 Simplex 64BitSupport ChecksumOffLoad
  DataRateSet

---------------------------------------------------------------
Statistics for adapters in the Shared Ethernet Adapter ent8
---------------------------------------------------------------
Number of adapters: 2
SEA Flags: 00000001
    < THREAD >
VLAN IDs :
    ent7: 1
Real Side Statistics:
    Packets received: 5701344
    Packets bridged: 5673198
    Packets consumed: 3963314
    Packets fragmented: 0
    Packets transmitted: 28685
    Packets dropped: 0
Virtual Side Statistics:
    Packets received: 0
    Packets bridged: 0
    Packets consumed: 0
    Packets fragmented: 0
    Packets transmitted: 5673253
    Packets dropped: 0
Other Statistics:
    Output packets generated: 28685
    Output packets dropped: 0
    Device output failures: 0
    Memory allocation failures: 0
    ICMP error packets sent: 0
    Non IP packets larger than MTU: 0
    Thread queue overflow packets: 0
High Availability Statistics:
    Control Channel PVID: 99
    Control Packets in: 0
    Control Packets out: 818825
Type of Packets Received:
    Keep-Alive Packets: 0
    Recovery Packets: 0
    Notify Packets: 0
    Limbo Packets: 0
    State: LIMBO
    Bridge Mode: All
    Number of Times Server became Backup: 0
    Number of Times Server became Primary: 0
    High Availability Mode: Auto
    Priority: 1

---------------------------------------------------------------
Real Adapter: ent2

ETHERNET STATISTICS (ent2) :
Device Type: 10/100 Mbps Ethernet PCI Adapter II (1410ff01)
Hardware Address: 00:0d:60:0c:05:00
```

```
Transmit Statistics:                          Receive Statistics:
--------------------                          -------------------
Packets: 28684                                Packets: 5701362
Bytes: 3704108                                Bytes: 664049607
Interrupts: 3                                 Interrupts: 5523380
Transmit Errors: 0                            Receive Errors: 0
Packets Dropped: 0                            Packets Dropped: 0
                                              Bad Packets: 0

Max Packets on S/W Transmit Queue: 2
S/W Transmit Queue Overflow: 0
Current S/W+H/W Transmit Queue Length: 1

Broadcast Packets: 21                         Broadcast Packets: 3740225
Multicast Packets: 0                          Multicast Packets: 194986
No Carrier Sense: 0                           CRC Errors: 0
DMA Underrun: 0                               DMA Overrun: 0
Lost CTS Errors: 0                            Alignment Errors: 0
Max Collision Errors: 0                       No Resource Errors: 0
Late Collision Errors: 0                      Receive Collision Errors: 0
Deferred: 0                                   Packet Too Short Errors: 0
SQE Test: 0                                   Packet Too Long Errors: 0
Timeout Errors: 0                             Packets Discarded by Adapter: 0
Single Collision Count: 0                     Receiver Start Count: 0
Multiple Collision Count: 0
Current HW Transmit Queue Length: 1

General Statistics:
-------------------
No mbuf Errors: 0
Adapter Reset Count: 0
Adapter Data Rate: 200
Driver Flags: Up Broadcast Running
 Simplex Promiscuous AlternateAddress
 64BitSupport ChecksumOffload PrivateSegment LargeSend DataRateSet

10/100 Mbps Ethernet PCI Adapter II (1410ff01) Specific Statistics:
------------------------------------------------------------------------
Link Status: Up
Media Speed Selected: Auto negotiation
Media Speed Running: 100 Mbps Full Duplex
Receive Pool Buffer Size: 1024
No Receive Pool Buffer Errors: 0
Receive Buffer Too Small Errors: 0
Entries to transmit timeout routine: 0
Transmit IPsec packets: 0
Transmit IPsec packets dropped: 0
Receive IPsec packets: 0
Receive IPsec SA offload count: 0
Transmit Large Send packets: 0
Transmit Large Send packets dropped: 0
Packets with Transmit collisions:
  1 collisions: 0     6 collisions: 0     11 collisions: 0
  2 collisions: 0     7 collisions: 0     12 collisions: 0
  3 collisions: 0     8 collisions: 0     13 collisions: 0
  4 collisions: 0     9 collisions: 0     14 collisions: 0
  5 collisions: 0    10 collisions: 0     15 collisions: 0

--------------------------------------------------------------
Virtual Adapter: ent7

ETHERNET STATISTICS (ent7) :
Device Type: Virtual I/O Ethernet Adapter (l-lan)
Hardware Address: 8a:83:54:5b:4e:9a

Transmit Statistics:                          Receive Statistics:
```

```
--------------------                          -------------------
Packets: 7949318                              Packets: 0
Bytes: 915447641                              Bytes: 0
Interrupts: 0                                 Interrupts: 0
Transmit Errors: 0                            Receive Errors: 0
Packets Dropped: 0                            Packets Dropped: 0
                                              Bad Packets: 0

Max Packets on S/W Transmit Queue: 0
S/W Transmit Queue Overflow: 0
Current S/W+H/W Transmit Queue Length: 0

Broadcast Packets: 5312065                    Broadcast Packets: 0
Multicast Packets: 265589                     Multicast Packets: 0
No Carrier Sense: 0                           CRC Errors: 0
DMA Underrun: 0                               DMA Overrun: 0
Lost CTS Errors: 0                            Alignment Errors: 0
Max Collision Errors: 0                       No Resource Errors: 0
Late Collision Errors: 0                      Receive Collision Errors: 0
Deferred: 0                                   Packet Too Short Errors: 0
SQE Test: 0                                   Packet Too Long Errors: 0
Timeout Errors: 0                             Packets Discarded by Adapter: 0
Single Collision Count: 0                     Receiver Start Count: 0
Multiple Collision Count: 0
Current HW Transmit Queue Length: 0

General Statistics:
-------------------
No mbuf Errors: 0
Adapter Reset Count: 0
Adapter Data Rate: 20000
Driver Flags: Up Broadcast Running
 Simplex Promiscuous AllMulticast
 64BitSupport ChecksumOffload DataRateSet

Virtual I/O Ethernet Adapter (l-lan) Specific Statistics:
---------------------------------------------------------
RQ Lingth: 4481
No Copy Buffers: 0
Trunk Adapter: True
  Priority: 1  Active: True
Filter MCast Mode: False
Filters: 255
  Enabled: 1  Queued: 0  Overflow: 0
LAN State: Operational

Hypervisor Send Failures: 2371664
  Receiver Failures: 2371664
  Send Errors: 0

Hypervisor Receive Failures: 0

ILLAN Attributes: 0000000000003103 [0000000000003103]

PVID:  1      VIDs: None

Switch ID: ETHERNET0

Buffers   Reg  Alloc  Min  Max   MaxA  LowReg
 tiny     512  512    512  2048  512   512
 small    512  512    512  2048  512   512
 medium   128  128    128  256   128   128
 large    24   24     24   64    24    24
 huge     24   24     24   64    24    24

--------------------------------------------------------------
Control Adapter: ent9
```

```
ETHERNET STATISTICS (ent9) :
Device Type: Virtual I/O Ethernet Adapter (l-lan)
Hardware Address: 8a:83:54:5b:4e:9b

Transmit Statistics:                          Receive Statistics:
--------------------                          -------------------
Packets: 821297                               Packets: 0
Bytes: 21353722                               Bytes: 0
Interrupts: 0                                 Interrupts: 0
Transmit Errors: 0                            Receive Errors: 0
Packets Dropped: 0                            Packets Dropped: 0
                                              Bad Packets: 0

Max Packets on S/W Transmit Queue: 0
S/W Transmit Queue Overflow: 0
Current S/W+H/W Transmit Queue Length: 0

Broadcast Packets: 821297                     Broadcast Packets: 0
Multicast Packets: 0                          Multicast Packets: 0
No Carrier Sense: 0                           CRC Errors: 0
DMA Underrun: 0                               DMA Overrun: 0
Lost CTS Errors: 0                            Alignment Errors: 0
Max Collision Errors: 0                       No Resource Errors: 0
Late Collision Errors: 0                      Receive Collision Errors: 0
Deferred: 0                                   Packet Too Short Errors: 0
SQE Test: 0                                   Packet Too Long Errors: 0
Timeout Errors: 0                             Packets Discarded by Adapter: 0
Single Collision Count: 0                     Receiver Start Count: 0
Multiple Collision Count: 0
Current HW Transmit Queue Length: 0

General Statistics:
-------------------
No mbuf Errors: 0
Adapter Reset Count: 0
Adapter Data Rate: 20000
Driver Flags: Up Broadcast Running
   Simplex 64BitSupport ChecksumOffload DataRateSet

Virtual I/O Ethernet Adapter (l-lan) Specific Statistics:
---------------------------------------------------------
RQ Length: 4481
No Copy Buffers: 0
Trunk Adapter: False
Filter MCast Mode: False
Filters: 255
  Enabled: 0  Queued: 0  Overflow: 0
LAN State: Operational

Hypervisor Send Failures: 0
  Receiver Failures: 0
  Send Errors: 0

Hypervisor Receive Failures: 0

ILLAN Attributes: 0000000000003002 [0000000000003002]

PVID:  99     VIDs:  None

Switch ID: ETHERNET0

Buffers        Reg  Alloc    Min     Max   MaxA  LowReg
 tiny          512    512    512    2048    512     512
 small         512    512    512    2048    512     512
```

```
medium       128    128    128     256    128     128
large         24     24     24      64     24      24
huge          24     24     24      64     24      24
```

Shared Ethernet Adapter statistics

Learn about general Shared Ethernet Adapter statistics, such as VLAN IDs and packet information, and view examples.

Statistic descriptions

Table 52. Descriptions of Shared Ethernet Adapter statistics

Statistic	Description
Number of adapters	Includes the real adapter and all of the virtual adapters. **Note:** If you are using Shared Ethernet Adapter failover, then the control channel adapter is not included.
Shared Ethernet Adapter flags	Denotes the features that the Shared Ethernet Adapter is currently running. **THREAD** The Shared Ethernet Adapter is operating in threaded mode, where incoming packets are queued and processed by different threads; its absence denotes interrupt mode, where packets are processed in the same interrupt where they are received. **LARGESEND** The large send feature has been enabled on the Shared Ethernet Adapter. **JUMBO_FRAMES** The jumbo frames feature has been enabled on the Shared Ethernet Adapter. **GVRP** The GVRP feature has been enabled on the Shared Ethernet Adapter.
VLAN IDs	List of VLAN IDs that have access to the network through the Shared Ethernet Adapter (this includes PVID and all tagged VLANs).

Table 52. Descriptions of Shared Ethernet Adapter statistics (continued)

Statistic	Description
Real adapters	**Packets received** Number of packets received on the physical network. **Packets bridged** Number of packets received on the physical network that were sent to the virtual network. **Packets consumed** Number of packets received on the physical network that were addressed to the interface configured over the Shared Ethernet Adapter. **Packets fragmented** Number of packets received on the physical network that were fragmented before being sent to the virtual network. They were fragmented because they were bigger than the outgoing adapter's Maximum Transmission Unit (MTU). **Packets transmitted** Number of packets sent on the physical network. This includes packets sent from the interface configured over the Shared Ethernet Adapter, as well as each packet sent from the virtual network to the physical network (including fragments). **Packets dropped** Number of packets received on the physical network that were dropped for one of the following reasons: • The packet was the oldest packet on a thread's queue and there was no space to accommodate a newly received packet. • The packet had an invalid VLAN ID and could not be processed. • The packet was addressed to the Shared Ethernet Adapter interface, but its interface had no filters registered.

Table 52. Descriptions of Shared Ethernet Adapter statistics (continued)

Statistic	**Description**
Virtual adapters	**Packets received** Number of packets received on the virtual network. In other words, the number of packets received on all of the virtual adapters. **Packets bridged** Number of packets received on the virtual network that were sent to the physical network. **Packets consumed** Number of packets received on the virtual network that were addressed to the interface configured over the Shared Ethernet Adapter. **Packets fragmented** Number of packets received on the virtual network that were fragmented before being sent to the physical network. They were fragmented because they were bigger than the outgoing adapter's MTU. **Packets transmitted** Number of packets sent on the virtual network. This includes packets sent from the interface configured over the Shared Ethernet Adapter, as well as each packet sent from the physical network to the virtual network (including fragments). **Packets dropped** Number of packets received on the virtual network that were dropped for one of the following reasons: • The packet was the oldest packet on a thread's queue and there was no space to accommodate a newly received packet. • The packet was addressed to the Shared Ethernet Adapter interface, but its interface had no filters registered.
Output packets generated	Number of packets with a valid VLAN tag or no VLAN tag sent out of the interface configured over the Shared Ethernet Adapter.
Output packets dropped	Number of packets sent out of the interface configured over the Shared Ethernet Adapter that are dropped because of an invalid VLAN tag.
Device output failures	Number of packets that could not be sent due to underlying device errors. This includes errors sent on the physical network and virtual network, including fragments and Internet Control Message Protocol (ICMP) error packets generated by the Shared Ethernet Adapter.
Memory allocation failures	Number of packets that could not be sent because there was insufficient network memory to complete an operation.

Table 52. Descriptions of Shared Ethernet Adapter statistics (continued)

Statistic	Description
ICMP error packets sent	Number of ICMP error packets successfully sent when a big packet could not be fragmented because the *don't fragment* bit was set.
Non IP packets larger than MTU	Number of packets that could not be sent because they were bigger than the outgoing adapter's MTU and could not be fragmented because they were not IP packets.
Thread queue overflow packets	Number of packets that were dropped from the thread queues because there was no space to accommodate a newly received packet.

Example statistics

```
ETHERNET STATISTICS (ent8) :
Device Type: Shared Ethernet Adapter
Hardware Address: 00:0d:60:0c:05:00
Elapsed Time: 3 days 20 hours 34 minutes 26 seconds

Transmit Statistics:                          Receive Statistics:
--------------------                          -------------------
Packets: 7978002                              Packets: 5701362
Bytes: 919151749                              Bytes: 664049607
Interrupts: 3                                 Interrupts: 5523380
Transmit Errors: 0                            Receive Errors: 0
Packets Dropped: 0                            Packets Dropped: 0
                                              Bad Packets: 0
Max Packets on S/W Transmit Queue: 2
S/W Transmit Queue Overflow: 0
Current S/W+H/W Transmit Queue Length: 1

Elapsed Time: 0 days 0 hours 0 minutes 0 seconds
Broadcast Packets: 5312086                    Broadcast Packets: 3740225
Multicast Packets: 265589                     Multicast Packets: 194986
No Carrier Sense: 0                           CRC Errors: 0
DMA Underrun: 0                               DMA Overrun: 0
Lost CTS Errors: 0                            Alignment Errors: 0
Max Collision Errors: 0                       No Resource Errors: 0
Late Collision Errors: 0                      Receive Collision Errors: 0
Deferred: 0                                   Packet Too Short Errors: 0
SQE Test: 0                                   Packet Too Long Errors: 0
Timeout Errors: 0                             Packets Discarded by Adapter: 0
Single Collision Count: 0                     Receiver Start Count: 0
Multiple Collision Count: 0
Current HW Transmit Queue Length: 1

General Statistics:
-------------------
No mbuf Errors: 0
Adapter Reset Count: 0
Adapter Data Rate: 0
Driver Flags: Up Broadcast Running
 Simplex 64BitSupport ChecksumOffLoad
  DataRateSet

--------------------------------------------------------------
Statistics for adapters in the Shared Ethernet Adapter ent8
--------------------------------------------------------------
Number of adapters: 2
SEA Flags: 00000001
    < THREAD >
VLAN IDs :
    ent7: 1
```

```
Real Side Statistics:
    Packets received: 5701344
    Packets bridged: 5673198
    Packets consumed: 3963314
    Packets fragmented: 0
    Packets transmitted: 28685
    Packets dropped: 0
Virtual Side Statistics:
    Packets received: 0
    Packets bridged: 0
    Packets consumed: 0
    Packets fragmented: 0
    Packets transmitted: 5673253
    Packets dropped: 0
Other Statistics:
    Output packets generated: 28685
    Output packets dropped: 0
    Device output failures: 0
    Memory allocation failures: 0
    ICMP error packets sent: 0
    Non IP packets larger than MTU: 0
    Thread queue overflow packets: 0

--------------------------------------------------------------
Real Adapter: ent2

ETHERNET STATISTICS (ent2) :
Device Type: 10/100 Mbps Ethernet PCI Adapter II (1410ff01)
Hardware Address: 00:0d:60:0c:05:00

Transmit Statistics:                          Receive Statistics:
--------------------                          -------------------
Packets: 28684                                Packets: 5701362
Bytes: 3704108                                Bytes: 664049607
Interrupts: 3                                 Interrupts: 5523380
Transmit Errors: 0                            Receive Errors: 0
Packets Dropped: 0                            Packets Dropped: 0
                                              Bad Packets: 0

Max Packets on S/W Transmit Queue: 2
S/W Transmit Queue Overflow: 0
Current S/W+H/W Transmit Queue Length: 1

Broadcast Packets: 21                         Broadcast Packets: 3740225
Multicast Packets: 0                          Multicast Packets: 194986
No Carrier Sense: 0                           CRC Errors: 0
DMA Underrun: 0                               DMA Overrun: 0
Lost CTS Errors: 0                            Alignment Errors: 0
Max Collision Errors: 0                       No Resource Errors: 0
Late Collision Errors: 0                      Receive Collision Errors: 0
Deferred: 0                                   Packet Too Short Errors: 0
SQE Test: 0                                   Packet Too Long Errors: 0
Timeout Errors: 0                             Packets Discarded by Adapter: 0
Single Collision Count: 0                     Receiver Start Count: 0
Multiple Collision Count: 0
Current HW Transmit Queue Length: 1

General Statistics:
-------------------
No mbuf Errors: 0
Adapter Reset Count: 0
Adapter Data Rate: 200
Driver Flags: Up Broadcast Running
 Simplex Promiscuous AlternateAddress
 64BitSupport ChecksumOffload PrivateSegment LargeSend DataRateSet

10/100 Mbps Ethernet PCI Adapter II (1410ff01) Specific Statistics:
```

```
------------------------------------------------------------------------
Link Status: Up
Media Speed Selected: Auto negotiation
Media Speed Running: 100 Mbps Full Duplex
Receive Pool Buffer Size: 1024
No Receive Pool Buffer Errors: 0
Receive Buffer Too Small Errors: 0
Entries to transmit timeout routine: 0
Transmit IPsec packets: 0
Transmit IPsec packets dropped: 0
Receive IPsec packets: 0
Receive IPsec SA offload count: 0
Transmit Large Send packets: 0
Transmit Large Send packets dropped: 0
Packets with Transmit collisions:
  1 collisions: 0     6 collisions: 0     11 collisions: 0
  2 collisions: 0     7 collisions: 0     12 collisions: 0
  3 collisions: 0     8 collisions: 0     13 collisions: 0
  4 collisions: 0     9 collisions: 0     14 collisions: 0
  5 collisions: 0    10 collisions: 0     15 collisions: 0

--------------------------------------------------------------
Virtual Adapter: ent7

ETHERNET STATISTICS (ent7) :
Device Type: Virtual I/O Ethernet Adapter (l-lan)
Hardware Address: 8a:83:54:5b:4e:9a

Transmit Statistics:                          Receive Statistics:
--------------------                          -------------------
Packets: 7949318                              Packets: 0
Bytes: 915447641                              Bytes: 0
Interrupts: 0                                 Interrupts: 0
Transmit Errors: 0                            Receive Errors: 0
Packets Dropped: 0                            Packets Dropped: 0
                                              Bad Packets: 0

Max Packets on S/W Transmit Queue: 0
S/W Transmit Queue Overflow: 0
Current S/W+H/W Transmit Queue Length: 0

Broadcast Packets: 5312065                    Broadcast Packets: 0
Multicast Packets: 265589                     Multicast Packets: 0
No Carrier Sense: 0                           CRC Errors: 0
DMA Underrun: 0                               DMA Overrun: 0
Lost CTS Errors: 0                            Alignment Errors: 0
Max Collision Errors: 0                       No Resource Errors: 0
Late Collision Errors: 0                      Receive Collision Errors: 0
Deferred: 0                                   Packet Too Short Errors: 0
SQE Test: 0                                   Packet Too Long Errors: 0
Timeout Errors: 0                             Packets Discarded by Adapter: 0
Single Collision Count: 0                     Receiver Start Count: 0
Multiple Collision Count: 0
Current HW Transmit Queue Length: 0

General Statistics:
-------------------
No mbuf Errors: 0
Adapter Reset Count: 0
Adapter Data Rate: 20000
Driver Flags: Up Broadcast Running
 Simplex Promiscuous AllMulticast
 64BitSupport ChecksumOffload DataRateSet

Virtual I/O Ethernet Adapter (l-lan) Specific Statistics:
---------------------------------------------------------
RQ Lingth: 4481
```

```
No Copy Buffers: 0
Trunk Adapter: True
  Priority: 1  Active: True
Filter MCast Mode: False
Filters: 255
  Enabled: 1  Queued: 0  Overflow: 0
LAN State: Operational

Hypervisor Send Failures: 2371664
  Receiver Failures: 2371664
  Send Errors: 0

Hypervisor Receive Failures: 0

ILLAN Attributes: 0000000000003103 [0000000000003103]

PVID:  1      VIDs: None

Switch ID: ETHERNET0

Buffers   Reg  Alloc  Min  Max   MaxA  LowReg
 tiny     512   512   512  2048  512   512
 small    512   512   512  2048  512   512
 medium   128   128   128  256   128   128
 large    24    24    24   64    24    24
 huge     24    24    24   64    24    24
```

User types for the Virtual I/O Server

Learn about Virtual I/O Server user types and their user permissions.

The Virtual I/O Server has the following user types: prime administrator, system administrator, service representative user, and development engineer user. After installation, the only user type that is active is the prime administrator.

Prime administrator

The prime administrator (**padmin**) user ID is the only user ID that is enabled after installation of the Virtual I/O Server and can run every Virtual I/O Server command. There can be only one prime administrator in the Virtual I/O Server.

System administrator

The system administrator user ID has access to all commands except the following commands:

- lsfailedlogin
- lsgcl
- mirrorios
- mkuser
- oem_setup_env
- rmuser
- shutdown
- unmirrorios

The prime administrator can create an unlimited number of system administrator IDs.

Service representative

Create the service representative (SR) user so that an IBM service representative can log in to the system and perform diagnostic routines. Upon logging in, the SR user is placed directly into the diagnostic menus.

Development engineer

Create a Development engineer (DE) user ID so that an IBM development engineer can log in to the system and debug problems.

View

This role is a read-only role and can perform only list-type (ls) functions. Users with this role do not have the authority to change the system configuration and do not have write permission to their home directories.

Notices

This information was developed for products and services offered in the U.S.A.

The manufacturer may not offer the products, services, or features discussed in this document in other countries. Consult the manufacturer's representative for information on the products and services currently available in your area. Any reference to the manufacturer's product, program, or service is not intended to state or imply that only that product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any intellectual property right of the manufacturer may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any product, program, or service.

The manufacturer may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to the manufacturer.

For license inquiries regarding double-byte character set (DBCS) information, contact the Intellectual Property Department in your country or send inquiries, in writing, to the manufacturer.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: THIS INFORMATION IS PROVIDED "AS IS " WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. The manufacturer may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to Web sites not owned by the manufacturer are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this product and use of those Web sites is at your own risk.

The manufacturer may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact the manufacturer.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, IBM License Agreement for Machine Code, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning products not produced by this manufacturer was obtained from the suppliers of those products, their published announcements or other publicly available sources. This manufacturer has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to products not produced by this manufacturer. Questions on the capabilities of products not produced by this manufacturer should be addressed to the suppliers of those products.

All statements regarding the manufacturer's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

The manufacturer's prices shown are the manufacturer's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to the manufacturer, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. The manufacturer, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information in softcopy, the photographs and color illustrations may not appear.

Programming interface information

This (ADD NAME OF PUBLICATION HERE) publication documents intended Programming Interfaces that allow the customer to write programs to obtain the services of (ADD PRODUCT NAME HERE).

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at Copyright and trademark information at www.ibm.com/legal/copytrade.shtml.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Red Hat, the Red Hat ″Shadow Man″ logo, and all Red Hat-based trademarks and logos are trademarks or registered trademarks of Red Hat, Inc., in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Terms and conditions

Permissions for the use of these publications is granted subject to the following terms and conditions.

Personal Use: You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative works of these publications, or any portion thereof, without the express consent of the manufacturer.

Commercial Use: You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of the manufacturer.

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any data, software or other intellectual property contained therein.

The manufacturer reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by the manufacturer, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

THE MANUFACTURER MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THESE PUBLICATIONS ARE PROVIDED ″AS-IS″ AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

REFERENCE
86 A1 81FA 01