Veritas Storage Foundation™ and High Availability Solutions Virtualization Guide

AIX

5.1 Service Pack 1

Symantec

Veritas Storage Foundation and High Availability Solutions Virtualization Guide

Product version: 5.1 SP1

Document version: 5.1SP1.1

Legal Notice

Symantec Corporation
350 Ellis Street
Mountain View, CA 94043

http://www.symantec.com

Technical Support

Symantec Technical Support maintains support centers globally. Technical Support's primary role is to respond to specific queries about product features and functionality. The Technical Support group also creates content for our online Knowledge Base. The Technical Support group works collaboratively with the other functional areas within Symantec to answer your questions in a timely fashion. For example, the Technical Support group works with Product Engineering and Symantec Security Response to provide alerting services and virus definition updates.

Symantec's support offerings include the following:

- A range of support options that give you the flexibility to select the right amount of service for any size organization
- Telephone and/or Web-based support that provides rapid response and up-to-the-minute information
- Upgrade assurance that delivers software upgrades
- Global support purchased on a regional business hours or 24 hours a day, 7 days a week basis
- Premium service offerings that include Account Management Services

For information about Symantec's support offerings, you can visit our Web site at the following URL:

www.symantec.com/business/support/index.jsp

All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policy.

Contacting Technical Support

Customers with a current support agreement may access Technical Support information at the following URL:

www.symantec.com/business/support/contact_techsupp_static.jsp

Before contacting Technical Support, make sure you have satisfied the system requirements that are listed in your product documentation. Also, you should be at the computer on which the problem occurred, in case it is necessary to replicate the problem.

When you contact Technical Support, please have the following information available:

- Product release level

- Hardware information
- Available memory, disk space, and NIC information
- Operating system
- Version and patch level
- Network topology
- Router, gateway, and IP address information
- Problem description:
  - Error messages and log files
  - Troubleshooting that was performed before contacting Symantec
  - Recent software configuration changes and network changes

Licensing and registration

If your Symantec product requires registration or a license key, access our technical support Web page at the following URL:

www.symantec.com/business/support/

Customer service

Customer service information is available at the following URL:

www.symantec.com/business/support/

Customer Service is available to assist with non-technical questions, such as the following types of issues:

- Questions regarding product licensing or serialization
- Product registration updates, such as address or name changes
- General product information (features, language availability, local dealers)
- Latest information about product updates and upgrades
- Information about upgrade assurance and support contracts
- Information about the Symantec Buying Programs
- Advice about Symantec's technical support options
- Nontechnical presales questions
- Issues that are related to CD-ROMs or manuals

Support agreement resources

If you want to contact Symantec regarding an existing support agreement, please contact the support agreement administration team for your region as follows:

Asia-Pacific and Japan	customercare_apac@symantec.com
Europe, Middle-East, and Africa	semea@symantec.com
North America and Latin America	supportsolutions@symantec.com

Documentation

Your feedback on product documentation is important to us. Send suggestions for improvements and reports on errors or omissions. Include the title and document version (located on the second page), and chapter and section titles of the text on which you are reporting. Send feedback to:

docs@symantec.com

About Symantec Connect

Symantec Connect is the peer-to-peer technical community site for Symantec's enterprise customers. Participants can connect and share information with other product users, including creating forum posts, articles, videos, downloads, blogs and suggesting ideas, as well as interact with Symantec product teams and Technical Support. Content is rated by the community, and members receive reward points for their contributions.

http://www.symantec.com/connect/storage-management

Contents

Chapter 1

Introduction

This chapter includes the following topics:

Purpose

This document explains the use of Storage Foundation High Availability (SFHA) 5.1 SP1 with IBM Power virtualization. The document shows how the SFHA products fit into the configuration for each virtualization solution. Where applicable, this document provides case studies. The scope of the document is limited to SFHA 5.1 SP1. While much of the functionality discussed in this document existed in previously released versions of the product, all case studies and configuration options are based upon this release.

Target Audience

The target audience for this document is a Solutions Architect, Solutions Planner, or a senior systems Administrator. Prior knowledge of Storage Foundation and Veritas Cluster Server are a prerequisite.

About Storage Foundation and High Availability Solutions

This topic describes the Storage Foundation and High Availability Solutions:

- *Storage Foundation* - Veritas Storage Foundation provides easy-to-use online storage management, enables high availability of data, optimized I/O performance, and allows freedom of choice in storage hardware investments. Veritas Storage Foundation is the base storage management offering from Symantec. It includes Veritas File System (VxFS) and Veritas Volume Manager (VxVM). Both VxFS and VxVM include advanced features such as journaling file system, storage checkpoints, dynamic multi-pathing , off-host processing, volume snapshots, and tiered storage. Storage Foundation comes in three editions: *Basic*, *Standard* and *Enterprise*.

  Each targets different environments as described below:

  - *Storage Foundation Basic* - Storage Foundation Basic is the freeware version of Storage Foundation. Available as a free download, it is limited to a maximum of 2 CPU and 4 volumes and 4 file systems.
  - *Storage Foundation Standard* - Storage Foundation Standard is intended for SAN connected servers with high performance requirements and availability features, such as multiple paths to storage. This product is a minimum requirement for High Availability solutions.
  - *Storage Foundation Enterprise* - Storage Foundation Enterprise includes the entire feature set of both File System and Volume Manager. It is designed for servers with large SAN connectivity, where high performance, off-host processing and tiered storage are desired.

- *Storage Foundation Cluster File System* - Veritas Storage Foundation Cluster File System provides an integrated solution for shared file environments. The solution includes Veritas Cluster File System, Cluster Volume Manager and Veritas Cluster Server to help implement robust, manageable, and scalable shared file solutions. Veritas Cluster File System provides linear scalability for parallel applications and is widely used as a fast failover mechanism to ensure that application downtime is minimized in the event of server or software failure. With Veritas Storage Foundation Cluster File System, cluster-wide volume and file system configuration allows for simplified management; and extending clusters are simplified as new servers adopt cluster-wide configurations.
- *Veritas Cluster Server* - Veritas Cluster Server is the industry's leading cross-platform clustering solution for minimizing application downtime. Through central management tools, automated failover, features to test disaster

recovery plans without disruption, and advanced failover management based on server capacity, Veritas Cluster Server allows IT managers to maximize resources by moving beyond reactive recovery to proactive management of application availability in heterogeneous environments.

Partition Mobility and Workload Migration

Additional information on IBM PowerVM configuration can be found in the *IBM Redbook* located at:

http://www.redbooks.ibm.com/redbooks/pdfs/sg247940.pdf

About IBM mLPARS with dedicated I/O

This is the baseline configuration.

Traditional AIX deployment with dedicated HBAs and NICs. Fully functional. This does not exclude partitions with virtual CPUs or partitions that support DLPAR events.

About Active Memory Sharing

The Veritas Storage Foundation High Availability stack supports VIO clients that use memory from the Active Memory Sharing (AMS) pool. Active Memory Sharing is a virtualization technology that allows multiple partitions to share a pool of physical memory. AMS increases system memory utilization and reduces the amount of physical memory that the system requires.

Symantec recommends that the ratio of the physical memory in the AMS pool should comply with the AIX guidelines.

See the IBM Redpaper PowerVM Virtualization Active Memory Sharing document for the AIX guidelines.

Supported Storage Foundation and High Availability Solutions functionality

In this release, Storage Foundation and High Availability products are fully functional in the IBM virtualization environment. The following functionality is supported:

- Veritas Cluster Server
  See "Veritas Cluster Server" on page 14.

- Veritas Storage Foundation
  See "Storage Foundation" on page 14.
- Storage Foundation Cluster File System, Storage Foundation for Oracle RAC, and Storage Foundation High Availability
  See "Storage Foundation Cluster File System, Storage Foundation for Oracle RAC, and Storage Foundation High Availability " on page 14.

Veritas Cluster Server

Veritas Cluster Server is support in the IBM virtualization environment. In this configuration, LLT/GAB run on physical devices, not on virtual NICs. Use individual NICs or active/backup (etherchannel) for CDN.

Storage Foundation

Storage Foundation stack is fully functional, including support for Portable Data Containers (Cross-platform Data Sharing format).

In the IBM virtualization environment, Storage Foundation runs on physical HBAs, and traditional LUN presentation is in place. Full DMP functionality is supported; the ability to query device information allows for LUN attribute discovery. This includes ALUA functionality, as well as thin reclamation and automated SSD discovery.

Storage Foundation Cluster File System, Storage Foundation for Oracle RAC, and Storage Foundation High Availability

Storage Foundation Cluster File System, Storage Foundation for Oracle RAC, and Storage Foundation High Availability are fully function in the IBM virtualization environment.

In this release, I/O fencing is supported. DMP is fully functional and supported.

Chapter 2

Veritas Cluster Server Solutions for IBM mLPARs with Virtual Ethernet

This chapter includes the following topics:

About IBM Virtual Ethernet

Virtual Ethernet enables communication between inter-partitions on the same server, without requiring each partition to have a physical network adapter. You can define in-memory connections between partitions that are handled at the system level (for example, interaction between POWER Hypervisor and the operating systems). These connections exhibit characteristics similar to physical high-bandwidth Ethernet connections and support the industry standard protocols (such as IPv4, IPv6, ICMP, or ARP). Virtual Ethernet also enables multiple partitions to share physical adapters for access to external networks using Shared Ethernet Adapter (SEA).

Shared Ethernet Adapter (SEA)

A Shared Ethernet Adapter is a layer-2 network bridge to securely transport network traffic between virtual Ethernet networks and physical network adapters. The SEA also enables several client partitions to share one physical adapter. The SEA is hosted in the Virtual I/O Server.

To bridge network traffic between the internal virtual network and external networks, configure the Virtual I/O Server with at least one physical Ethernet adapter. Multiple virtual Ethernet adapters can share one SEA. Each virtual Ethernet adapter can support multiple VLANs.

The SEA has the following characteristics:

- Virtual Ethernet MAC addresses of virtual Ethernet adapters are visible to outside systems (using the `arp -a` command).
- Supports unicast, broadcast, and multicast. Protocols such as Address Resolution Protocol (ARP), Dynamic Host Configuration Protocol (DHCP), Boot Protocol (BOOTP), and Neighbor Discovery Protocol (NDP) can work across an SEA.

VCS configuration in the Virtual Ethernet environment

To use VCS in the Virtual Ethernet environment, configure VCS according to the following sections:

- Configure the LLT private links.
  See "LLT Private links configuration " on page 16.
- Configure the VCS Agents.
  See "VCS Agents" on page 17.

LLT Private links configuration

LLT heartbeats

LLT uses standard Ethernet networks to provide communication for its heartbeats. These networks can be provided through physical ports or virtual Ethernet interfaces. These interfaces do not require IP addresses to be configured since LLT heartbeats are based on layer 2 protocols. The best practice includes two independent paths for heartbeats to eliminate any single point of failure. This scenario includes redundant VIO servers with each providing a virtual Ethernet to each client LPAR participating in the VCS cluster.

LLT Private Links connections

The diagrams illustrate LLT Heartbeat connections in an IBM VIO environment with Virtual Ethernet, Shared Ethernet Adapters, and external LPARs. The three node cluster consists of (2) VIO Client Partitions in System A and (1) LPAR in System B. POWER6 based systems that are controlled by the same Hardware Management Console (HMC).

MTU settings

Virtual Ethernet allows fairly large MTU for communication between LPARs. Communication through the Shared Ethernet is limited to much smaller MTU supported by the physical media. Therefore, choose the MTU for the Virtual Ethernet such that packets can be sent outside using the Shared Ethernet without any packet drop. You must make sure that LLT configuration file has MTU=1500 set for each of the virtual Ethernet interface you use for the private links.

The VCS installer detects the virtual Ethernet interfaces and sets the correct MTU in the LLT configuration file. If you are installing with manual steps, you must configure the MTU before you start the LLT.

Sample output of the /etc/llttab file restricting the MTU size to 1500:

```
# more /etc/llttab
set-node vcs_node_2
set-cluster 1234
link en1 /dev/dlpi/en:1 - ether - 1500
link en2 /dev/dlpi/en:2 - ether - 1500
```

After you configure the LLT, use the below command on all the nodes of your cluster to be sure that the overall MTU size is less than 1500.

```
# lltstat -c | grep mtu
mtu: 1460
```

VCS Agents

E.g. The NIC agent might need to be tuned to handle Virtual Ethernet failures. If there is no such tuning needed for any agent then we can say that all the VCS agents work in Virtual Ethernet environment without any tuning specific to this environment.

Virtual Ethernet and Cluster Management Software

Virtual Ethernet environment offers various advantages and flexibility, but you should be aware of the challenges. The various independent clusters consisting of VIO client partitions in the same physical computer can be configured with the heartbeat routed through the same physical Ethernet adapters to additional nodes outside the physical computer. Ensure that each cluster has a unique cluster ID. Unique cluster IDs eliminate conflict and allow the Virtual Ethernet environment to greatly reduce the required number of physical Ethernet adapters. According to IBM, there are issues to be aware that are not the fault of the applicable Cluster

Management Software or the configuration. Rather, the issues arise as a direct consequence of I/O virtualization.

To reiterate, although some of these may be viewed as configuration restrictions, many are direct consequences of I/O Virtualization.

The issues and recommendation are as follows:

- If two or more Clustered nodes use a VIO server or servers in the same frame, the Cluster Management Software cannot detect and react to single physical interface failures. This behavior does not limit the availability of the entire cluster because VIOS itself routes traffic around the failure. The behavior of the VIOS is analogous to AIX the EtherChannel. Notification of individual Adapter failures must use other methods (not based on the VIO server) .
- All Virtual Ethernet interfaces that are defined to the Cluster Management Software should be treated as “single-Adapter networks” according to IBM. To correctly monitor and detect failure of the network interface, you must create a file that includes a list of clients to ping. Due to the nature of Virtual Ethernet, other mechanisms to detect the failure of network interfaces are not effective.
- If the VIO server has only a single physical interface on a network, then the Cluster Management Software can detect a failure of that interface. However, that failure isolates the node from the network.

Check the IBM documentation for detailed information on the Virtual Ethernet and various configuration scenarios using virtual I/O Server. For information about the above issues, see the following link:

http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/FLASH10390

Chapter 3

Storage Foundation and High Availability Virtualization Solutions for IBM mLPARs with Virtual SCSI Devices

This chapter includes the following topics:

About IBM mLPARs with virtual SCSI devices

This discussion of vSCSI devices applies only to SAN-based LUNs presented through VIO. Internal devices, volumes, and files presented by VIO as vSCSI devices are not recommended for use with Storage Foundation.

Virtual SCSI uses a client/server model. A Virtual I/O server partition owns the physical I/O devices, and exports the devices as virtual SCSI (vSCSI) resources to the client partitions. The Virtual I/O client is a logical partition that has a virtual client adapter node defined in its device tree. The VIO client uses the vSCSI resources provided by the Virtual I/O Server partition to access the block interface devices.

If redundant SAN connections exist to the VIO server, the VIO server provides multi-pathing to the array. Client partitions can also perform multi-pathing

between VIO servers in an active/standby configuration. This configuration provides extended protection from VIO configuration and maintenance. Redundant VIO servers are recommended for production workloads.

What is a virtual SCSI (vSCSI) disk?

A virtual SCSI (vSCSI) disk is a resource which can be a SCSI disk, or a volume or file in a VIO Server that is exported to a virtual IO client. IBM vSCSI LUNs implement a sub-set of the SCSI protocol. The two main limitations are:

- Persistent reservations (SCSI3 – PGR) are not implemented.
  The lack of SCSI reservations means that I/O Fencing is not supported. Storage Foundation Cluster File System (SFCFS) and Storage Foundation for Oracle RAC (SFRAC) do not support vSCSI disks, because SFCFS and SFRAC require I/O fencing.
- Device inquiry limitations.
  Veritas Storage Foundation (SF) cannot directly fetch the inquiry data, as is done from a physical SCSI disk. Because of this limitation, releases before SF 5.1 do not support cross-platform data sharing (CDS) functionality for vSCSI disks.
  Starting with release SF 5.1, CDS functionality is supported.

Using Storage Foundation in the VIO client with virtual SCSI devices

Storage Foundation provides support for virtual SCSI (vSCSI) devices on the VIO client. You can create and manage Veritas Volume Manager (VxVM) volumes on vSCSI devices, as for any other devices. Storage Foundation provides Dynamic Multi-Pathing (DMP) for vSCSI devices, by default. Storage Foundation can also co-exist with MPIO for multi-pathing. If you choose to use MPIO to multipath the vSCSI devices, DMP works in pass-through mode.

Use the `vxddladm` utility and the `vxdmpadm` utility to administer DMP for vSCSI devices. The `vxddladm` utility controls enabling and disabling DMP on vSCSI devices, adding and removing supported arrays, and listing supported arrays. The `vxdmpadm` utility controls the I/O policy and the path policy for vSCSI devices.

Using Storage Foundation with virtual SCSI devices

Versions of SF that support vSCSI disks are:

Prior to Storage Foundation 5.1, Portable Data Containers (disk type CDS) were not supported. With extensions included in Storage Foundation 5.1, CDS type devices are now supported.

Storage Foundation can be used in the following ways:

- use DMP in the VIO server to provide multi-pathing to the array. DMP presents a dmpnode as a vSCSI device to the VIO client.
- use Storage Foundation in the VIO client to provide volume management on the vSCSI devices, and multi-pathing through the VIO servers with DMP.
- use SF in the VIO client to provide volume management on the vSCSI devices, and use MPIO to provide multi-pathing.

Setting up DMP for vSCSI devices in the Virtual I/O Client

In this release of Storage Foundation, Veritas Dynamic Multi-Pathing (DMP) is enabled on VIO clients by default. After you install or upgrade Storage Foundation in the Virtual IO client, any vSCSI devices are under DMP control. MPIO is disabled.

If you have already installed or upgraded Storage Foundation in the Virtual I/O client, use the following procedure to enable DMP support for vSCSI devices. This procedure is only required if you have previously disabled DMP support for vSCSI devices.

To enable vSCSI support within DMP and disable MPIO

1 Enable vSCSI support.

```
# vxddladm enablevscsi
```

2 You are prompted to reboot the devices, if required.

DMP takes control of the devices, for any array that has DMP support to use the array for vSCSI devices. You can add or remove DMP support for vSCSI for arrays.

See "Adding and removing DMP support for vSCSI devices for an array" on page 23.

About disabling DMP multi-pathing for vSCSI devices in the Virtual IO Client

Storage Foundation can co-exist with MPIO multi-pathing in the Virtual I/O client. If you prefer to use MPIO for multi-pathing, you can override the default behavior, which enables Dynamic Multi-Pathing (DMP) in the Virtual I/O client.

There are two ways to do this:

- Before you install or upgrade Storage Foundation in the Virtual I/O client

See "Preparing to install or upgrade Storage Foundation with DMP disabled for vSCSI devices in the Virtual I/O client" on page 22.

- After Storage Foundation is installed in the Virtual I/O client
  See "Disabling DMP multi-pathing for vSCSI devices in the Virtual IO Client, after installation" on page 22.

Preparing to install or upgrade Storage Foundation with DMP disabled for vSCSI devices in the Virtual I/O client

Before you install or upgrade Storage Foundation, you can set an environment variable to disable DMP use for the vSCSI devices. Storage Foundation is installed with DMP in pass-through mode. MPIO is enabled for multi-pathing.

Note: When you upgrade an existing VxVM installation that has DMP enabled, then DMP remains enabled regardless of whether or not the environment variable __VXVM_DMP_VSCSI_ENABLE is set to no.

To disable DMP before installing or upgrading SF in the Virtual I/O Client

1. Before you install or upgrade VxVM, set the environment variable __VXVM_DMP_VSCSI_ENABLE to no.

   ```
   # export __VXVM_DMP_VSCSI_ENABLE=no
   ```

   Note: The environment variable name __VXVM_DMP_VSCSI_ENABLE begins with two underscore (_) characters.

   ---

2. Install Storage Foundation, as described in the *Storage Foundation High Availability Installation Guide*

Disabling DMP multi-pathing for vSCSI devices in the Virtual IO Client, after installation

After VxVM is installed, use the `vxddladm` command to switch vSCSI devices between MPIO control and DMP control.

To return control to MPIO, disable vSCSI support with DMP. After DMP support has been disabled, MPIO takes control of the devices. MPIO implements multi-pathing features such as failover and load balancing; DMP acts in pass-through mode.

To disable vSCSI support within DMP and enable MPIO

1 Disable vSCSI support.

```
# vxddladm disablevscsi
```

2 You are prompted to reboot the devices, if required.

Adding and removing DMP support for vSCSI devices for an array

Veritas Dynamic Multi-Pathing (DMP) controls the devices for any array that has DMP support to use the array for vSCSI devices.

To add or remove DMP support for an array for use with vSCSI devices

1 To determine if DMP support is enabled for an array, list all of the arrays that DMP supports for use with vSCSI devices:

```
# vxddladm listvscsi
```

2 If the support is not enabled, add support for using an array as a vSCSI device within DMP:

```
# vxddladm addvscsi array_vid
```

3 If the support is enabled, you can remove the support so that the array is not used for vSCSI devices within DMP:

```
# vxddladm rmvscsi array_vid
```

4 You are prompted to reboot the system, if required.

How DMP handles I/O for vSCSI devices

On the VIO client, DMP uses the Active/Standby array mode for the vSCSI devices. Each path to the vSCSI device is through a VIO server. One VIO server is Active and the other VIO servers are Standby. An Active/Standby array permits I/O through a single Active path, and keeps the other paths on standby. During failover, I/O is scheduled on one of the standby paths. After failback, I/Os are scheduled back onto the original Active path. The Active/Standby mode is a variation of an active/active array; only one path is active at a time.

The DMP I/O policy for vSCSI devices is always Single-Active. You cannot change the DMP I/O policy for the vSCSI enclosure. Because only one VIO server can be Active, DMP cannot do I/O balancing across the paths for vSCSI devices.

The following command shows the vSCSI enclosure:

```
# vxdmpadm listenclosure all
ENCLR_NAME   ENCLR_TYPE  ENCLR_SNO   STATUS     ARRAY_TYPE  LUN_COUNT
=============================================================================
ibm_vscsi0   IBM_VSCSI   VSCSI       CONNECTED  VSCSI   9
```

The following command shows the I/O policy for the vSCSI enclosure:

```
# vxdmpadm getattr enclosure ibm_vscsi0 iopolicy
ENCLR_NAME     DEFAULT        CURRENT
============================================
ibm_vscsi0     Single-Active  Single-Active
```

For vSCSI devices, DMP balances the load between the VIO servers, instead of balancing the I/O on paths. By default, the `iopolicy` attribute of the vSCSI enclosure is set to `lunbalance`. When `lunbalance` is set, the vSCSI LUNs are distributed so that the I/O load is shared across the VIO servers. For example, if you have 10 LUNs and 2 VIO servers, 5 of them are configured so that VIO Server 1 is Active and VIO Server 2 is Standby. The other 5 are configuredd so that the VIO Server 2 is Active and VIO Server 1 is Standby. To turn off load sharing across VIO servers, set the `iopolicy` attribute to `nolunbalance`.

DMP dynamically balances the I/O load across LUNs. When you add or remove disks or paths in the VIO client, the load is rebalanced. Temporary failures like enabling or disabling paths or controllers do not cause the I/O load across LUNs to be rebalanced.

Setting the vSCSI I/O policy

By default, DMP balances the I/O load across VIO servers. This behavior sets the I/O policy attribute to `lunbalance`.

To display the current I/O policy attribute for the vSCSI array

- Display the current I/O policy for a vSCSI array:

```
# vxdmpadm getattr vscsi iopolicy
VSCSI     DEFAULT      CURRENT
============================================
IOPolicy  lunbalance   lunbalance
```

To turn off the LUN balancing, set the I/O policy attribute for the vSCSI array to `nolunbalance`.

To set the I/O policy attribute for the vSCSI array

- Set the I/O policy for a vSCSI array:

```
# vxdmpadm setattr vscsi iopolicy={lunbalance|nolunbalance}
```

Note: The DMP I/O policy for each vSCSI device is always Single-Active. You cannot change the DMP I/O policy for the vSCSI enclosure. Only one VIO server can be Active for each vSCSI device.

Chapter 4

Veritas Dynamic Multi-Pathing for the Virtual I/O Server

This chapter includes the following topics:

Virtual I/O server overview

Virtual I/O (VIO) server is a virtualization technology by IBM. A Virtual I/O server is a logical partition (LPAR) that runs a trimmed-down version of the AIX operating system. Virtual I/O servers have APV support, which allows sharing of physical I/O resources between virtual I/O clients.

See the PowerVM wiki for more in-depth information about VIO server and virtualization:

http://www.ibm.com/developerworks/wikis/display/virtualization/VIO

For more information, see the *PowerVM Virtualization on IBM System p redbook*:

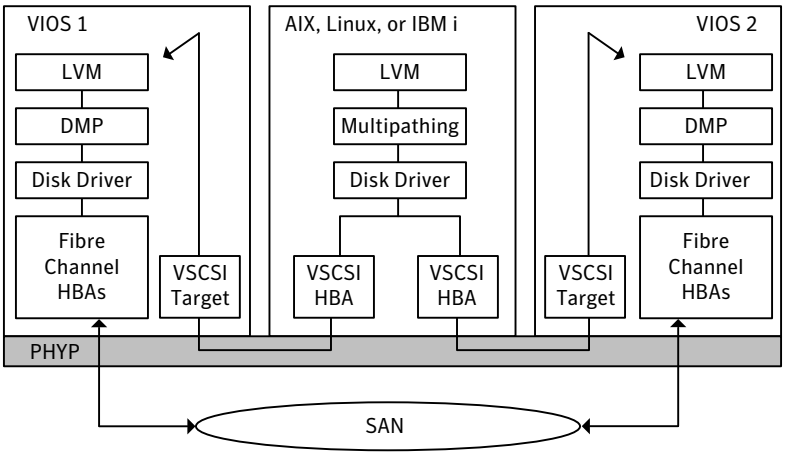
http://www.redbooks.ibm.com/redpieces/abstracts/sg247940.html

DMP support for Virtual I/O Server

DMP support in Virtual I/O Server requires a Veritas Dynamic Multi-Pathing (DMP) product license. Minimum VIOS oslevel required is 2.1.3 10-FP-23 or later.

Figure 4-1 illustrates DMP enablement in the Virtual I/O Server.

Figure 4-1 Veritas Dynamic Multi-Pathing in the Virtual I/O Server



DMP administration and management on Virtual I/O Server

DMP is fully functional in the Virtual I/O server. DMP administration and management commands (`vxdmpadm`, `vxddladm`, `vxdisk`, etc.) must be invoked from the non-restricted root shell.

```
$ oem_setup_env
```

Some example commands:

```
dmpvios1$ vxdmpadm getsubpaths dmpnodename=ibm_ds8x000_0337

NAME     STATE[A] PATH-TYPE[M] CTLR-NAME ENCLR-TYPE ENCLR-NAME  ATTRS
================================================================================
hdisk21 ENABLED(A)  -          fscsi0    IBM_DS8x00  ibm_ds8x000  -
hdisk61 ENABLED(A)  -          fscsi0    IBM_DS8x00  ibm_ds8x000  -
```

```
hdisk80 ENABLED(A)  -         fscsi1   IBM_DS8x00  ibm_ds8x000  -
hdisk99 ENABLED(A)  -         fscsi1   IBM_DS8x00  ibm_ds8x000  -

dmpvios1$ vxdmpadm listenclosure all

ENCLR_NAME  ENCLR_TYPE ENCLR_SNO STATUS    ARRAY_TYPE LUN_COUNT
============================================================
disk        Disk       DISKS     CONNECTED Disk       1
ibm_ds8x000 IBM_DS8x00 75MA641   CONNECTED A/A        6
```

For complete information about managing Dynamic Multi-Pathing, see the *Veritas Dynamic Multi-Pathing Administrator's Guide.*

Veritas Volume Manager (VxVM) administration and management

Veritas Volume Manager (VxVM) functionality is disabled in Virtual I/O Server. VxVM commands that manage volumes or disk groups are disabled in the VIO server.

In the VIOS, VxVM does not detect disk format information, so the disk status for VxVM disks is shown as unknown. For example:

```
dmpvios1$ vxdisk list
DEVICE           TYPE     DISK    GROUP   STATUS
disk_0           auto     -       -       unknown
ibm_ds8x000_02c1 auto     -       -       unknown
ibm_ds8x000_0288 auto     -       -       unknown
ibm_ds8x000_029a auto     -       -       unknown
ibm_ds8x000_0292 auto     -       -       unknown
ibm_ds8x000_0293 auto     -       -       unknown
ibm_ds8x000_0337 auto     -       -       unknown
```

In the VIOS, VxVM displays an error if you run a command that is disabled, as follows:

```
dmpvios1$ vxdisk -f init ibm_ds8x000_0288
VxVM vxdisk ERROR V-5-1-5433 Device ibm_ds8x000_0288: init failed:
Operation not allowed. VxVM is disabled.

dmpvios1$ vxdg import bootdg
VxVM vxdg ERROR V-5-1-10978 Disk group bootdg: import failed:
Operation not allowed. VxVM is disabled.
```

Configuring DMP on Virtual I/O Server

In this release, you can install DMP in the virtual I/O server (VIOS). This enables the VIO server to export dmpnodes to the VIO clients. The VIO clients access the dmpnodes in the same way as any other vSCSI devices. DMP handles the I/O to the disks backed by the dmpnodes.

Installing Veritas Dynamic Multi-Pathing (DMP) on Virtual I/O Server

Veritas Dynamic Multi-Pathing (DMP) can operate in the Virtual I/O server. Install DMP on the Virtual I/O server.

To install DMP on the Virtual I/O Server

1. Log into the VIO server partition.
2. Use the `oem_setup_env` command to access the non-restricted root shell.
3. Install Veritas Dynamic Multi-Pathing on the Virtual I/O Server.

   See the *Veritas Dynamic Multi-Pathing Installation Guide*.
4. Installing DMP on the VIO server enables the dmp_native_support tunable. Do not set the dmp_native_support tunable to off.

```
dmpvios1$ vxdmpadm gettune dmp_native_support
Tunable                Current Value  Default Value
------------------  --------------- -------------------
dmp_native_support        on              off
```

Migrating from other multi-pathing solutions to DMP on Virtual I/O Server

DMP supports migrating from AIX MPIO and EMC PowerPath multi-pathing solutions to DMP on Virtual I/O Server.

To migrate from other multi-pathing solutions to DMP on Virtual I/O Server

1. Before migrating, back up the Virtual I/O Servers to use for reverting the system in case of issues.
2. Shut down all VIO client partitions that are serviced by the VIOS.
3. Log into the VIO server partition. Use the following command to access the non-restricted root shell. All subsequent commands in this procedure must be invoked from the non-restricted shell.

```
$ oem_setup_env
```

4 For each Fibre Channel (FC) adapter on the system, verify that the following attributes have the recommended settings:

fc_err_recov	fast_fail
dyntrk	yes

If required, use the `chdev` command to change the attributes.

The following example shows how to change the attributes:

```
dmpvios1$ chdev -a fc_err_recov=fast_fail -a dyntrk=yes -l \
 fscsi0 -P
fscsi0 changed
```

The following example shows the new attribute values:

```
dmpvios1$ lsattr -El  fscsi0


attach       switch   How this adapter is CONNECTED  False
dyntrk       yes      Dynamic Tracking of FC Devices True
fc_err_recov fast_fail FC Fabric Event Error RECOVERY
Policy True
scsi_id      0xd0c00  Adapter SCSI ID                False
sw_fc_class  3        FC Class for Fabric            True
```

5 Use commands like `lsdev` and `lsmap` to view the configuration.

6 Unconfigure all VTD devices from all virtual adapters on the system:

```
dmpvios1$ rmdev -p vhost0
```

Repeat this step for all other virtual adapters.

7 Migrate from the third-party device driver to DMP.

Note that you do not need to do turn on the dmp_native_support again, because it is turned on for VIOS by default. You can use the `vxdmpadm gettune dmp_native_support` command to verify that the tunable parameter is turned on.

For the migration procedure, see the *Veritas Dynamic Multi-Pathing Administrator's Guide*.

8 Reboot the VIO Server partition.

9 Use the following command to verify that all Virtual SCSI mappings of TPD multi-pathing solution have been correctly migrated to DMP:

```
dmpvios1$ /usr/ios/cli/ioscli lsmap -all
```

10 Repeat step 1 through step 9 for all of the other VIO server partitions of the managed system.

11 After all of the VIO Server partitions are successfully migrated to DMP, start all of the VIO client partitions.

Example: migration from MPIO to DMP on Virtual I/O Server for a dual-VIOS configuration

This section shows an example of a migration from MPIO to DMP on the Virtual I/O Server, in a configuration with two VIO Servers.

Example configuration:

```
Managed System:  dmpviosp6
VIO server1: dmpvios1
VIO server2: dmpvios2
VIO clients: dmpvioc1
SAN LUNs: IBM DS8K array
Current multi-pathing solution on VIO server: IBM MPIO

ODM definition fileset required to disable MPIO support
for IBM DS8K array LUNs:
devices.fcp.disk.ibm.rte
```

To migrate dmpviosp6 from MPIO to DMP

1 Before migrating, back up the Virtual I/O Server to use for reverting the system in case of issues.

See the IBM website for information about backing up Virtual I/O Server.

2 Shut down all of the VIO clients that are serviced by the VIO Server.

```
dmpvioc1$ halt
```

3 Log into the VIO server partition.Use the following command to access the non-restricted root shell. All subsequent commands in this procedure must be invoked from the non-restricted shell.

```
$ oem_setup_env
```

4 Verify that the FC adapters have the recommended settings. If not, change the settings as required.

For example, the following output shows the settings:

```
dmpvios1$ lsattr -El fscsi0
attach       switch  How this adapter is CONNECTED    False
dyntrk       yes     Dynamic Tracking of FC Devices   True
fc_err_recov fast_fail FC Fabric Event Error RECOVERY
Policy True
scsi_id      0xd0c00 Adapter SCSI ID                  False
sw_fc_class  3       FC Class for Fabric              True
```

5 The following command shows `lsmap` output before migrating MPIO VTD devices to DMP:

```
dmpvios1$ /usr/ios/cli/iosctl lsmap -all
SVSA            Physloc                      Client Partition ID
--------------- ---------------------------- ------------------
vhost0          U9117.MMA.0686502-V2-C11     0x00000004

VTD               vtscsi0
Status            Available 8100000000000000
Backing device    hdisk21
LUN               0x
Physloc           U789D.001.DQD04AF-P1-C5-T1-W500507630813861A-L4
003403700000000

VTD               vtscsi1
Status            Available
LUN               0x8200000000000000
Backing device    hdisk20
Physloc           U789D.001.DQD04AF-P1-C5-T1-W500507630813861A-L4
00240C100000000

VTD               vtscsi2
Status            Available
LUN               0x8300000000000000
Backing device    hdisk18
Physloc           U789D.001.DQD04AF-P1-C5-T1-W500507630813861A-L4
002409A00000000
```

The VIO Server has MPIO providing multi-pathing to these hdisks. The following commands show the configuration:

```
dmpvios1$ lsdev -Cc disk | egrep "hdisk21|hdisk20|hdisk18"

hdisk18 Available 02-08-02 MPIO Other FC SCSI Disk Drive
hdisk20 Available 02-08-02 MPIO Other FC SCSI Disk Drive
hdisk21 Available 02-08-02 MPIO Other FC SCSI Disk Drive
```

6 Unconfigure all VTD devices from all virtual adapters on the system:

```
dmpvios1 $ rmdev -p vhost0
vtscsi0 Defined
vtscsi1 Defined
vtscsi2 Defined
```

Repeat this step for all other virtual adapters.

7 Migrate the devices from MPIO to DMP.

Unmount the file system and varyoff volume groups residing on the MPIO devices.

Display the volume groups (vgs) in the configuration:

```
dmpvios1$ lsvg
rootvg
brunovg

dmpvios1 lsvg -p brunovg

brunovg:
PV_NAME PV STATE TOTAL PPs FREE PPs FREE DISTRIBUTION
hdisk19 active   511       501      103..92..102..102..102
hdisk22 active   511       501      103..92..102..102..102
```

Use the `varyoffvg` command on all affected vgs:

```
dmpvios1$ varyoffvg brunovg
```

Install the IBMDS8K ODM definition fileset to remove IBM MPIO support for IBM DS8K array LUNs.

```
dmpvios1$ installp -aXd . devices.fcp.disk.ibm.rte


+-----------------------------------------------------+
             Pre-installation Verification...
+-----------------------------------------------------+
Verifying selections...done
Verifying requisites...done
Results...
Installation Summary
--------------------
Name                      Level   Part  Event  Result
-------------------------------------------------------
devices.fcp.disk.ibm.rte  1.0.0.2 USR   APPLY  SUCCESS
devices.fcp.disk.ibm.rte  1.0.0.2 ROOT  APPLY  SUCCESS
```

8 Reboot VIO server1

```
dmpvios1$ reboot
```

9 After the VIO server1 reboots, verify that all of the existing volume groups on the VIO server1 and MPIO VTDs on the VIO server1 are successfully migrated to DMP.

```
dmpvios1 lsvg -p brunovg

brunovg:
PV_NAME          PV STATE TOTAL PPs FREE PPs FREE DISTRIBUTION
ibm_ds8000_0292 active   511       501    103..92..102..102..102
ibm_ds8000_0293 active   511       501    103..92..102..102..102
```

Verify the vSCSI mappings of IBM DS8K LUNs on the migrated volume groups:

```
dmpvios1 lsmap -all

SVSA            Physloc                          Client Partition ID
--------------- -------------------------------- ------------------
vhost0          U9117.MMA.0686502-V2-C11     0x00000000
VTD                   vtscsi0
Status                Available
LUN                   0x8100000000000000
Backing device        ibm_ds8000_0337
Physloc

VTD                   vtscsi1
Status                Available
LUN                   0x8200000000000000
Backing device        ibm_ds8000_02c1
Physloc

VTD                   vtscsi2
Status                Available
LUN                   0x8300000000000000
Backing device        ibm_ds8000_029a
Physloc
```

10 Repeat step 1 through step 9 for VIO server2.

11 Start all of the VIO clients using HMC.

Example: migration from PowerPath to DMP on Virtual I/O Server for a dual-VIOS configuration

This section shows an example of a migration from PowerPath to DMP on the Virtual I/O Server, in a configuration with two VIO Servers.

Example configuration:

```
Managed System:  dmpviosp6
VIO server1: dmpvios1
VIO server2: dmpvios2
VIO clients: dmpvioc1
SAN LUNs: EMC Clariion array
Current multi-pathing solution on VIO server: EMC PowerPath
```

To migrate dmpviosp6 from PowerPath to DMP

1. Before migrating, back up the Virtual I/O Server to use for reverting the system in case of issues.

   See the IBM website for information about backing up Virtual I/O Server.

2. Shut down all of the VIO clients that are serviced by the VIO Server.

   ```
   dmpvioc1$ halt
   ```

3. Log into the VIO server partition.Use the following command to access the non-restricted root shell. All subsequent commands in this procedure must be invoked from the non-restricted shell.

   ```
   $ oem_setup_env
   ```

4. Verify that the FC adapters have the recommended settings. If not, change the settings as required.

   For example, the following output shows the settings:

   ```
   dmpvios1$ lsattr -El fscsi0
   attach       switch    How this adapter is CONNECTED  False
   dyntrk       yes       Dynamic Tracking of FC Devices True
   fc_err_recov fast_fail FC Fabric Event Error RECOVERY Policy
    True
   scsi_id      0xd0c00   Adapter SCSI ID                False
   sw_fc_class  3         FC Class for Fabric            True
   ```

5 The following command shows `lsmap` output before migrating PowerPath VTD devices to DMP:

```
dmpvios1$ /usr/ios/cli/iosctl lsmap -all


SVSA            Physloc                       Client Partition ID
--------------- ----------------------------- --------------------
vhost0          U9117.MMA.0686502-V2-C11      0x00000004

VTD                P0
Status             Available
LUN                0x8100000000000000
Backing device     hdiskpower0
Physloc            U789D.001.DQD04AF-P1-C5-T1-W500507630813861A-L4
0034037
00000000

VTD               P1
Status            Available
LUN               0x8200000000000000
Backing device    hdiskpower1
Physloc           U789D.001.DQD04AF-P1-C5-T1-W500507630813861A-L40
0240C10
0000000

VTD               P2
Status            Available
LUN               0x8300000000000000
Backing device    hdiskpower2
Physloc           U789D.001.DQD04AF-P1-C5-T1-W500507630813861A-L40
02409A00000000
```

6 Unconfigure all VTD devices from all virtual adapters on the system:

```
dmpvios1 $ rmdev -p vhost0
P0 Defined
P1 Defined
P2 Defined
```

Repeat this step for all other virtual adapters.

7 Migrate the devices from PowerPath to DMP.

Unmount the file system and varyoff volume groups residing on the PowerPath devices.

Display the volume groups (vgs) in the configuration:

```
dmpvios1$ lsvg
rootvg
brunovg

dmpvios1 lsvg -p brunovg

brunovg:
PV_NAME     PV STATE TOTAL PPs FREE PPs FREE DISTRIBUTION
hdiskpower3 active   511       501    103..92..102..102..102
```

Use the varyoffvg command on all affected vgs:

```
dmpvios1$ varyoffvg brunovg
```

Unmanage the EMC Clariion array from PowerPath control

```
# powermt unmanage class=clariion
hdiskpower0 deleted
hdiskpower1 deleted
hdiskpower2 deleted
hdiskpower3 deleted
```

8 Reboot VIO server1

```
dmpvios1$ reboot
```

9 After the VIO server1 reboots, verify that all of the existing volume groups on the VIO server1 and MPIO VTDs on the VIO server1 are successfully migrated to DMP.

```
dmpvios1 lsvg -p brunovg

brunovg:
PV_NAME      PV STATE TOTAL PPs FREE PPs FREE DISTRIBUTION
emc_clari0_138 active  511      501    103..92..102..102..102
```

Verify the mappings of the LUNs on the migrated volume groups:

```
dmpvios1 lsmap -all

SVSA            Physloc                     Client Partition ID
--------------- --------------------------- ------------------
vhost0          U9117.MMA.0686502-V2-C11    0x00000000
VTD                   P0
Status                Available
LUN                   0x8100000000000000
Backing device        emc_clari0_130
Physloc

VTD                   P1
Status                Available
LUN                   0x8200000000000000
Backing device        emc_clari0_136
Physloc

VTD                   P2
Status                Available
LUN                   0x8300000000000000
Backing device        emc_clari0_137
Physloc
```

10 Repeat step 1 to step 9 for VIO server2.

11 Start all of the VIO clients.

Configuring DMP pseudo devices as virtual SCSI devices

DMP in the VIO server supports the following methods to export a device to the VIO client:

- DMP node method
  See "Exporting DMP devices as Virtual SCSI disks " on page 42.
- Logical partition-based method
  See "Exporting a Logical Volume as a Virtual SCSI disk" on page 45.
- File-based method
  See "Exporting a file as a virtual SCSI disk" on page 47.

Exporting DMP devices as Virtual SCSI disks

DMP supports disks backed by DMP as Virtual SCSI disks. Export the DMP device as a vSCSI disk to the VIO client.

To export a DMP device as a vSCSI disk

1. Log into the VIO server partition.Use the following command to access the non-restricted root shell. All subsequent commands in this procedure must be invoked from the non-restricted shell.

   ```
   $ oem_setup_env
   ```

2. Use the `oem_setup_env` command to access the non-restricted root shell.
3. The following command displays the DMP devices on the VIO server:

   ```
   dmpvios1$ lsdev -t dmpdisk

   ibm_ds8000_0287 Available  Veritas DMP Device
   ibm_ds8000_0288 Available  Veritas DMP Device
   ibm_ds8000_0292 Available  Veritas DMP Device
   ibm_ds8000_0293 Available  Veritas DMP Device
   ibm_ds8000_029a Available  Veritas DMP Device
   ibm_ds8000_02c1 Available  Veritas DMP Device
   ibm_ds8000_0337 Available  Veritas DMP Device
   ```

4. Assign the DMP device as a backing device. Exit from the non-restricted shell to run this command from the VIOS default shell.

   ```
   dmpvios1$ exit

   $ mkvdev -vdev ibm_ds8000_0288 -vadapter vhost0
   vtscsi3 Available
   ```

5 Use the following command to display the configuration.

```
$ lsmap -all

SVSA            Physloc                          Client Partition ID
--------------- -------------------------------- ------------------
vhost0          U9117.MMA.0686502-V2-C11         0x00000000
VTD                   vtscsi0
Status                Available
LUN                   0x8100000000000000
Backing device        ibm_ds8000_0337
Physloc

VTD                   vtscsi1
Status                Available
LUN                   0x8200000000000000
Backing device        ibm_ds8000_02c1
Physloc

VTD                   vtscsi2
Status                Available
LUN                   0x8300000000000000
Backing device        ibm_ds8000_029a
Physloc V

TD                    vtscsi3
Status                Available
LUN                   0x8400000000000000
Backing device        ibm_ds8000_0288
Physloc
```

6 For a dual-VIOS configuration, export the DMP device corresponding to the same SAN LUN on the second VIO Server in the configuration. To export the DMP device on the second VIO server, identify the DMP device corresponding to the SAN LUN as on the VIO Server1.

- If the array supports the AVID attribute, the DMP device name is the same as the DMP device name on the VIO Server1.
- Otherwise, use the UDID value of the DMP device on the VIO Server1 to correlate the DMP device name with same UDID on the VIO Server2.
  On VIO Server1:

  ```
  $ oem_setup_env
  ```

```
dmpvios1$ lsattr -El ibm_ds8000_0288

attribute value            description        user_settable
dmpname   ibm_ds8x000_0288 DMP Device name    True
pvid      none             Physical volume identifier True
unique_id IBM%5F2107%5F75MA641%5F6005076308FFC61A000000000
0000288
Unique device identifier   True
```

On VIO Server2:

```
$ oem_setup_env

dmpvios2$ odmget -q "attribute = unique_id and
value = 'IBM%5F2107%5F75MA641%5F6005076308FFC61A000000000
0000288'" CuAt

CuAt:

   name = "ibm_ds8000_0288"
   attribute = "unique_id"
   value = "IBM%5F2107%5F75MA641%5F6005076308FFC61A00
   00000000000288"
   type = "R"
   generic = "DU"
   rep = "s"
   nls_index = 4
```

7 Use the DMP device name identified in step 6 to assign the DMP device as a backing device. Exit from the non-restricted shell to run this command from the VIOS default shell.

```
dmpvios1$ exit

$ mkvdev -vdev ibm_ds8000_0288 -vadapter vhost0
vtscsi3 Available
```

8 Use the following command to display the configuration.

```
$ lsmap -all

SVSA            Physloc                     Client Partition ID
--------------  --------------------------- ------------------
vhost0          U9117.MMA.0686502-V2-C11    0x00000000
VTD                   vtscsi0
Status                Available
LUN                   0x8100000000000000
Backing device        ibm_ds8000_0337
Physloc

VTD                   vtscsi1
Status                Available
LUN                   0x8200000000000000
Backing device        ibm_ds8000_02c1
Physloc

VTD                   vtscsi2
Status                Available
LUN                   0x8300000000000000
Backing device        ibm_ds8000_029a
Physloc V

TD                    vtscsi3
Status                Available
LUN                   0x8400000000000000
Backing device        ibm_ds8000_0288
Physloc
```

Exporting a Logical Volume as a Virtual SCSI disk

DMP supports vSCSI disks backed by a Logical Volume. Export the Logical Volume as a vSCSI disk to the VIO client.

To export a Logical Volume as a vSCSI disk

1 Create the volume group.

```
$ mkvg -vg brunovg ibm_ds8000_0292 ibm_ds8000_0293
brunovg
```

The following command displays the new volume group:

```
$ lsvg -pv brunovg
brunovg:
PV_NAME          PV STATE TOTAL PPs FREE PPs FREE DISTRIBUTION
ibm_ds8000_0292 active   494       494      99..99..98..99..99
ibm_ds8000_0293 active   494       494      99..99..98..99..99
```

2 Make a logical volume in the volume group.

```
$ mklv -lv brunovg_lv1 brunovg 1G
brunovg_lv1
```

The following command displays the new logical volume:

```
$ lsvg -lv brunovg
brunovg:
LV NAME       TYPE   LPs   PPs   PVs  LV STATE      MOUNT POINT
brunovg_lv1   jfs    256   256   1    closed/syncd  N/A
```

3 Assign the logical volume as a backing device.

```
$ mkvdev -vdev brunovg_lv1 -vadapter  vhost0
vtscsi4 Available
```

4 Use the following command to display the configuration.

```
$ lsmap -all

SVSA            Physloc                      Client Partition ID
--------------- ---------------------------- ------------------
vhost0          U9117.MMA.0686502-V2-C11     0x00000000
VTD                   vtscsi0
Status                Available
LUN                   0x8100000000000000
Backing device        ibm_ds8000_0337
Physloc

VTD                   vtscsi1
Status                Available
LUN                   0x8200000000000000
Backing device        ibm_ds8000_02c1
Physloc

VTD                   vtscsi2
Status                Available
LUN                   0x8300000000000000
Backing device        ibm_ds8000_029a
Physloc

VTD                   vtscsi3
Status                Available
LUN                   0x8400000000000000
Backing device        ibm_ds8000_0288
Physloc

VTD                   vtscsi4
Status                Available
LUN                   0x8500000000000000
Backing device        brunovg_lv1
Physloc
```

Exporting a file as a virtual SCSI disk

DMP supports vSCSI disks backed by a file. Export the file as a vSCSI disk to the VIO client.

To export a file as a vSCSI disk

1 Create the storage pool.

```
$ mksp brunospool ibm_ds8000_0296
brunospool
0516-1254 mkvg: Changing the PVID in the ODM.
```

2 Create a file system on the pool.

```
$ mksp -fb bruno_fb -sp brunospool -size 500M
bruno_fb
File system created successfully.
507684 kilobytes total disk space.
New File System size is 1024000
```

3 Mount the file system.

```
$ mount


node       mounted   mounted over   vfs    date     options
---------- ----------------------   ----- --------------------
/dev/hd4       /        jfs2      Jul 02 14:47 rw,log=/dev/hd8
/dev/hd2      /usr      jfs2      Jul 02 14:47 rw,log=/dev/hd8
/dev/hd9var   /var      jfs2      Jul 02 14:47 rw,log=/dev/hd8
/dev/hd3      /tmp      jfs2      Jul 02 14:47 rw,log=/dev/hd8
/dev/hd1      /home     jfs2      Jul 02 14:48 rw,log=/dev/hd8
/dev/hd11admin /admin   jfs2      Jul 02 14:48 rw,log=/dev/hd8
/proc         /proc     procfs    Jul 02 14:48 rw
/dev/hd10opt  /opt      jfs2      Jul 02 14:48 rw,log=/dev/hd8
/dev/livedump /var/adm/ras/livedump jfs2 Jul 02 14:48 rw,log=
/dev/hd8
/dev/bruno_fb /var/vio/storagepools/bruno_fb jfs2 Jul 02 15:38
rw,log=INLINE
```

4 Create a file in the storage pool.

```
$ mkbdsp -bd bruno_fbdev -sp bruno_fb 200M
Creating file "bruno_fbdev" in storage pool "bruno_fb".
bruno_fbdev
```

5 Assign the file as a backing device.

```
$ mkbdsp -sp bruno_fb -bd bruno_fbdev -vadapter vhost0
Assigning file "bruno_fbdev" as a backing device.
vtscsi5 Available
bruno_fbdev
```

6 Use the following command to display the configuration.

```
$ lsmap -all

SVSA            Physloc                      Client Partition ID
--------------- ---------------------------- ------------------
vhost0          U9117.MMA.0686502-V2-C11     0x00000000
...
...
VTD                vtscsi5
Status             Available
LUN                0x8600000000000000
Backing device     /var/vio/storagepools/bruno_fb/bruno_fbdev
Physloc
```

Extended attributes in VIO client for a Virtual SCSI disk

Using DMP in the Virtual I/O server enables the DMP in the VIO Client to receive the extended attributes for the LUN. This enables the client LPAR to view back-end LUN attributes such as thin, SSD, and RAID levels associated with the vSCSI devices.

For more information about extended attributes and the prerequisites for supporting them, see the following tech note:

http://seer.entsupport.symantec.com/docs/337516.htm

Configuration prerequisites for providing extended attributes on VIO client for Virtual SCSI disk

DMP in VIO client will provide extended attributes information of backend SAN LUN. The following conditions are prerequisites for using extended attributes on the VIO client:

- VIO client has vSCSI disks backed by SAN LUNs.
- In the VIO Server partition, DMP is controlling those SAN LUNs.
- On VIO client, DMP is controlling the vSCSI disks.

Displaying extended attributes of Virtual SCSI disks

When a VIO client accesses a virtual SCSI disk that is backed by a DMP device on the Virtual I/O Server, the VIO client can access the extended attributes associated with the virtual SCSI disk.

The following commands can access and display extended attributes information associated with the vSCSI disk backed by DMP device on Virtual I/O Server.

- `vxdisk -e list`
- `vxdmpadm list dmpnodename=<daname>`
- `vxdmpadm -v getdmpnode dmpnodename=<daname>`
- `vxdisk -p list <daname>`

For example, use the following command on the VIO client dmpvioc1:

```
# vxdisk -e list
```

```
DEVICE           TYPE          DISK   GROUP  STATUS       OS_NATIVE_NAME  ATTR
ibm_ds8x000_114f auto:LVM      -      -      LVM          hdisk83         std
3pardata0_3968   auto:aixdisk  -      -      online thin  hdisk84         tp
```

```
# vxdmpadm list dmpnode dmpnodename=3pardata0_3968
```

```
dmpdev          = 3pardata0_3968
state           = enabled
enclosure       = 3pardata0
cab-sno         = 744
asl             = libvxvscsi.so
vid             = AIX
pid             = VDASD
array-name      = 3PARDATA
array-type      = VSCSI
iopolicy        = Single-Active
avid            = 3968
lun-sno         = 3PARdata%5FVV%5F02E8%5F2AC00F8002E8
udid            = AIX%5FVDASD%5F%5F3PARdata%255FVV%255F02E8%255F2AC00F8002E8
dev-attr        = tp
###path         = name state type transport ctlr hwpath aportID aportWWN attr
path            = hdisk84 enabled(a) - SCSI vscsi1 vscsi1 3 - -
```

Chapter 5

Storage Foundation and High Availability Virtualization Solutions for IBM mLPARs with N_Port ID Virtualization

This chapter includes the following topics:

About IBM mLPARs with N_Port ID Virtualization (NPIV)

N_Port ID Virtualization or NPIV is a Fibre Channel industry standard technology that allows multiple N_Port IDs to share a single physical N_Port. NPIV provides the capability to take a single physical Fibre Channel HBA port and divide it such that it appears, to both the host and to the SAN, as though there are multiple World Wide Port Names (WWPNs).

NPIV provides direct access to the Fibre Channel adapters from multiple virtual machine (client partitions), simplifying zoning and storage allocation. Resources

can be zoned directly to the virtual client, which has its own World Wide Port Name (WWPN).

The use of NPIV with IBM VIO provides the capability to use a single Fibre Channel port and overlay multiple WWPNs so that it appears to the SAN as both the VIO server and client partitions. NPIV enables the AIX VIO server to provision entire dedicated logical ports to client mLPARs rather than individual LUNs. Client partitions with this type of logical port operates as though the partition has its own dedicated FC protocol adapter. To utilize the NPIV functionality, a new type of virtual Fibre Channel (VFC) adapter is defined on both the VIO and Client. A server VFC adapter can only be created on a VIO server partition; a client VFC adapter can only be created on client partitions. WWPNs are allocated to client VFC adapters when they are defined in the profile, based upon an assignment pool generated from the backing physical adapter.

There is always corresponding one-to-one mapping relationship between VFC adapters on client logical partitions and VFC on the VIOS. That is, each VFC that is assigned to a client logical partition must connect to only one VFC adapter on VIOS, and each VFC on VIOS must connect to only one VFC on the client logical partition.

NPIV support is included with PowerVM Express, Standard, and Enterprise Edition and supports AIX V5.3 and AIX V6.1.

More details on NPIV and how to configure IBM VIO environment is available in IBM documentation.

Characteristics of a LUN through NPIV

To the operating system, multi-pathing drivers and system tools, a LUN presented through NPIV has all the characteristics of a LUN presented through a dedicated HBA. Device inquiry and probing works as with physical HBAs. When a VFC interface is created, two WWNs are assigned. This information is available in the HMC as part of the virtual HBA properties.

All SCSC device inquiry operations work, allowing for array identification functions, visibility of LUN Device Identifiers, and discovery of such attributes as thin and thin re-claim capability. SCSI-3 persistent reservation functionality is also supported, enabling the use of SCSI-3 I/O Fencing if the underlying storage supports.

When Zoning/LUN mapping operations occur, care should be made to ensure that storage is assigned to both WWNs. During normal operation, only one of the WWN identifiers is in use, but during a Live Partition migration event, the WWN identifier not previously used will be configured on the appropriate backing HBA on the target system, log into the SAN, and then become the active WWN. The

previously used WWN will become inactive until the next Live Partition Mobility operation.

VIO requirements

NPIV requires Power6 systems, VIOS 2.1, and 8GB HBA adapters (model number xxxxx). NPIV also requires NPIV aware switches. The end storage devices need not be NPIV aware.

Hardware requirements

NPIV requires extended functionality on the HBA. Currently IBM sells this as an 8GB HBA, part number XXXXX. The SAN Switch ports must also support NPIV as well, Brocade and Cisco make products that provide this functionality.

Support for Storage Foundation in NPIV environment

Starting with Storage Foundation 5.0MP3 RP1, Storage Foundation supports the IBM Virtual I/O Server (VIOS) environment. The VIOS is configured with NPIV capable FC adapters that are connected to a SAN switch that is NPIV capable. The LUNs mapped to the VIO client behave like an LPAR having a dedicated FC adapter. The devices in the VIO client appear as regular SCSI disks. Storage Foundation can access these LUNS, and treat these devices as if they came from a regular SAN storage array LUN. Unlike in the classic VIO environment without NPIV, SF treats these devices as if they came from a regular SAN storage array LUN. With NPIV, the VIO client environment is transparent to SF. All of the SF commands would have the same output as in a regular physical AIX server. SF identifies the vSCSI LUNs through the array properties of the LUNs. Otherwise, the devices in the VIO client appear as regular SCSI disks. You can import the disk group, which provides access to volumes and file systems.

Appendix A lists the details of the environment and configuration, including the firmware versions that are used during the NPIV support qualification with SF 5.0MP3 RP1 (AIX).

Symantec has qualified NPIV support with SF, starting with 5.0MP3 RP1.

Storage Foundation

Storage Foundation 5.1 SP1 supports all functionality available with dedicated HBAs when using LUNs presented through NPIV. All IBM supported NPIV enabled HBAs are supported by Storage Foundation.

Storage Foundation functionality is fully supported with NPIV.

Cluster File System

Cluster File System is supported with NPIV.

Installation, patching, and configuration requirements

Symantec strongly recommends that you use Storage Foundation 5.1 SP1 with the latest patches. No other configuration is required. Refer to the following website for the latest patches for Storage Foundation 5.1 SP1 on AIX:

https://vos.symantec.com/checklist/install/

Chapter 6

Storage Foundation support for Live Partition Mobility

This chapter includes the following topics:

About Live Partition Mobility (LPM)

The Live Partition Mobility available on POWER6 based systems enables you to migrate an entire logical partition from one system to another. Live Partition Mobility transfers the configuration from source to destination without disrupting the hosted applications or the setup of the operating system and applications. Live Partition Mobility gives you a greater control over the usage of resources in the data center.

It allows a level of reconfiguration that in the past was not possible due to complexity or because of service level agreements that do not allow an application to be stopped for an architectural change. The migration process can be performed in the following ways:

- Inactive migration
  The logical partition is powered off and moved to the destination system.
- Active migration

The migration of the partition is performed while service is provided, without disrupting user activities. During an active migration, the applications continue to handle their normal workload. Disk data transactions, running network connections, user contexts, and the complete environment are migrated without any loss and migration can be activated any time on any production partition.

SFHA supported configuration

All SFHA stacks support LPM including Fencing configured with NPIV disks.

Note: Please check the IBM documentation for the detailed information on the LPM requirements and LPM process.

Requirements for the Live Partition Mobility

The main requirements for the migration of a logical partition are:

Two POWER6 based systems controlled by the same Hardware Management Console (HMC). The destination system must have enough CPU and memory resources to host the mobile partition.

Network requirements: The migrating partition must use the virtual LAN for all LLT links and public network access. The VLAN must be bridged (if there is more than one, then it also has to be bridged) to a physical network using a shared Ethernet adapter in the Virtual I/O Server partition. The Virtual I/O Servers on both systems must have a shared Ethernet adapter configured to bridge to the same Ethernet network used by the mobile partition. Your LAN must be configured such that migrating partitions can continue to communicate with the other nodes after a migration is completed.

Storage requirements: The operating system, applications, and data of the mobile partition must reside on virtual storage on an external storage subsystem since the mobile partition's disk data must be available after the migration to the destination system is completed. An external, shared access storage subsystem is required. The mobile partition's virtual disks must be mapped to LUNs; they cannot be part of a storage pool or logical volume on the Virtual I/O Server. The LUNs must be zoned and masked to the Virtual I/O Servers on both systems.

Overview of partition migration process

The partition migration, either inactive or active, is divided into the following stages:

- Preparing the infrastructure to support Live Partition Mobility.

- Checking the configuration and readiness of the source and destination systems.
- Transferring the partition state from the source to destination. The same command is used to launch inactive and active migrations. The HMC determines the appropriate type of migration to use based on the state of the mobile partition.
- Completing the migration by freeing unused resources on the source system and the HMC.

Performance considerations

Active partition migration involves moving the state of a partition from one system to another while the partition is still running. The mover service partitions working with the hypervisor use partition virtual memory functions to track changes to partition memory state on the source system while it is transferring memory state to the destination system. During the migration phase, there is an initial transfer of the mobile partition's physical memory from the source to the destination. Since the mobile partition is still active, a portion of the partition's resident memory will almost certainly have changed during this pass. The hypervisor keeps track of these changed pages for retransmission to the destination system in a dirty page list. It makes additional passes through the changed pages until the mover service partition detects that a sufficient amount of pages are clean or the timeout is reached. The speed and load of the network used to transfer state between the source and destination systems influence the time required for both the transfer of the partition state and the performance of any remote paging operations. The amount of changed resident memory after the first pass is controlled more by write activity of the hosted applications than by the total partition memory size. Nevertheless, it is reasonable to assume that partitions with a large memory requirement will have higher numbers of changed resident pages than smaller ones. To ensure that active partition migrations are truly non-disruptive, even for large partitions, the POWER Hypervisor resumes the partition on the destination system before all the dirty pages have been migrated over to the destination. If the mobile partition tries to access a dirty page that has not yet been migrated from the source system, the hypervisor on the destination sends a demand paging request to the hypervisor on the source to fetch the required page. Providing a high-performance network between the source and destination mover partitions and reducing the partition's memory update activity prior to migration will improve the latency of the state transfer phase of migration. We suggest using a dedicated network for state transfer, with a nominal bandwidth of at least 1 Gbps.

Chapter 7

Storage Foundation support for IBM Workload Partitions

This chapter includes the following topics:

About IBM Workload Partitions

IBM Workload Partitions (WPARs) are implemented within AIX 6.1. Workload Partitions allow administrators to virtualize the AIX operating system, by partitioning an AIX operating system instance into multiple environments. Each environment within the AIX operating system instance is called a workload partition (WPAR). One WPAR can host applications and isolate the applications from applications executing in other WPARs. WPAR is a pure software solution and has no dependencies on hardware features.

The WPAR solution allows for fewer operating system images on your IBM System p partitioned server. Prior to WPARs, you had to create a new Logical Partition (LPAR) for each new "isolated" environment. With AIX 6.1, you can instead use multiple WPARs within one LPAR, in many circumstances.

In an LPAR environment, each LPAR requires its own operating system image and a certain number of physical resources. While you can virtualize many of these resources, some physical resources must be allocated to the system for each LPAR. Furthermore, you need to install patches and technology upgrades to each LPAR. Each LPAR requires its own archiving strategy and DR strategy. It also

takes some time to create an LPAR; you also need to do this outside of AIX, through a Hardware Management Console (HMC) or the Integrated Virtualization Manager (IVM).

In contrast, WPARs are much simpler to manage and can be created from the AIX command line or through SMIT.WPARs allow you to avoid the biggest disadvantage of LPARs: maintaining multiple images, and therefore possibly over-committing expensive hardware resources, such as CPU and RAM. While partitioning helps you consolidate and virtualize hardware within a single box, operating system virtualization through WPAR technology goes one step further and allows for an even more granular approach of resource management.

The WPAR solution shares operating system images and is clearly the most efficient use of CPU, RAM, and I/O resources. Rather than a replacement for LPARs, WPARs are a complement to them and allow one to further virtualize application workloads through operating system virtualization. WPARs allow for new applications to be deployed much more quickly.

On the other hand, it's important to understand the limitations of WPARs. For example, each LPAR is a single point of failure for all WPARs that are created within the LPAR. In the event of an LPAR problem (or a scheduled system outage), all underlying WPARs are also affected.

The following sections describe the types of WPARs:

- System workload partition
- Application workload partition

System workload partition

The system WPAR is much closer to a complete version of AIX. The system WPAR has its own dedicated, completely writable file-systems along with its own inetd and cron. You can define remote access to the System workload partition.

Application workload partition

Application WPARs are lightweight versions of virtualized OS environments. They are extremely limited and can only run application processes, not system daemons such as inetd or cron. You cannot even define remote access to this environment. These are only temporarily objects; they actually disintegrate when the final process of the application partition ends, and as such, are more geared to execute processes than entire applications. WPARs have no real dependency on hardware and can even be used on POWER4 systems that do not support IBM's PowerVM (formerly known as APV). For AIX administrators, the huge advantage

of WPARs is the flexibility of creating new environments without having to create and manage new AIX partitions.

When to use WPARs

Use WPARs when you need an isolated environment and you do not want to create new LPARs because of the limitation of the available resources. Here are the few scenarios:

- Application/workload isolation
- Quickly testing an application

WPARs share the global resources with other WPARs in the same LPAR, which limits the usefulness of WPARs in some situations.

We recommend not using WPARs in the following situations:

- Security: WPAR processes can be seen by the global environment from the central LPAR. If you are running a highly secure type of system, this may be a problem for you from a security standpoint. Further, the root administrator of your LPAR will now have access to your workload partition, possibly compromising the security that the application may require.
- Performance: Each WPAR within the LPAR uses the same system resources of the LPAR. You need to be more careful when architecting your system and also when stress testing the system.
- Availability: If you are in an environment where it is very difficult to bring a system down, it's important to note that when performing maintenance on an LPAR that every WPAR defined will be affected. At the same time, if there is a system panic and AIX crashes, every WPAR has now been brought down.
- Production: Avoid using WPARs in the production environment. LPARs provide more granularity and complete OS isolation.
- Physical devices: Physical devices are not supported within a WPAR. More details on WPAR administration can be found in the IBM red book on WPARs at
  http://www.redbooks.ibm.com/abstracts/sg247431.html

Storage Foundation support for WPARs

This section describes Storage Foundation support for WPARs.

Veritas File System support as namefs in WPAR (SF 5.1 release)

This section describes Veritas File System (VxFS) support for WPAR The Veritas Storage Foundation File System (local mount only) is supported inside the workload partition (WPAR) environment only through NFS mount options. Cluster mount is not yet supported inside a WPAR.

WPAR with VxFS for non root partition

In Storage Foundation release 5.1, there is limited support for WPARs, as follows:

- All the Storage Foundation packages must be installed and configured in the global partition of AIX.
- Storage Foundation can only be administered from the global partition.
- Using NFS mount, the storage resources of Storage Foundation, primarily file system mount points, can be used to create and configure system or application WPARs.

Prior to 5.1SP1 VXFS was supported as namefs in WPAR. The following procedure describes how this support was achieved.

To use VxFS as namefs in the WPAR (prior to release 5.1SP1)

1 Create a vxfs filesystem in the global environment:

```
/opt/VRTS/mkfs -V vxfs /dev/vx/rdsk/testvg/vol1
```

2 Create a WPAR. For example, use the following command.

```
mkwpar -n devpayrollWPAR01
```

For other options while creating WPARs, refer to the IBM Redbook for WLPAR.

3 List the WPAR.

```
# lswpar
Name                State  Type  Hostname  Directory
-------------------------------------------------------------------
devpayrollWPAR01  D  S     devpayrollWPAR01 /wpars/devpayrollWPAR01
```

4 The above output shows that WPAR does not have the devices. To get the vxfs file system in WPAR, to create the file system in the global environment. Then mount it to the WPAR directories which are located at /wpar/<wparname>/

```
# mkdir /wpars/devpayrollWPAR01/vxfs_dir
#  mount -V vxfs /dev/vx/dsk/testdg/vol1/wpars/devpayrollWPAR01/vxfs_dir
```

5 Start the WPAR:

```
# startwpar -Dv fsqawpar
2>/startwpar_tl2
```

6 Log in to the WPAR.

```
# clogin hostname
```

For example, to log in to the WPAR devpayrollWPAR01:

```
# clogin devpayrollWPAR01
```

7 The following output shows the VxFS mount point in the WPAR as namefs.

```
# mount
node       mounted      mounted over vfs   date         options
-------- -------------- ------------ ----- ------------ ------------
/dev/vx/dsk/testdg/vol1 /vxfs_dir     vxfs Jun 23 03:14 rw,delaylog,
suid,ioerror=mwdisable,qio,largefiles
/dev/fslv01      /               jfs2   Jun 23 03:15 rw,log=INLINE
/dev/fslv02      /home           jfs2   Jun 23 03:15 rw,log=INLINE
/opt             /opt            namefs Jun 23 03:15 ro
/proc            /proc           namefs Jun 23 03:15 rw
/dev/fslv03      /tmp            jfs2   Jun 23 03:15 rw,log=INLINE
/usr             /usr            namefs Jun 23 03:15 ro
/dev/fslv04      /var            jfs2   Jun 23 03:15 rw,log=INLINE
```

8 To stop the WPAR:

```
# stopwpar -Dv devpayrollWPAR01
2>/devpayrollWPAR01_tl2
```

9 If we mount vxfs on some directory in Global environment and then mount that directory in /wpar/ devpayrollWPAR01/vxfs_dir.

After login to devpayrollWPAR01 /vxfs_dir will appear as namefs.

```
# mount -V vxfs /dev/vx/dsk/testdg/vol1  /mnt

# mount /mnt /wpars/devpayrollWPAR01/vxfs_dir/

# startwpar -Dv devpayrollWPAR01
2>/devpayrollWPAR01_tl2

# clogin devpayrollWPAR01
# mount
node       mounted        mounted over    vfs       date        options
------- ---------------  ---------------  ------ ------------ ---------------
/mnt             /vxfs_dir        namefs Jun 23 03:29 rw
/dev/fslv01      /                jfs2   Jun 23 03:30 rw,log=INLINE
/dev/fslv02      /home            jfs2   Jun 23 03:30 rw,log=INLINE
/opt             /opt             namefs Jun 23 03:30 ro
/proc            /proc            namefs Jun 23 03:30 rw
/dev/fslv03      /tmp             jfs2   Jun 23 03:30 rw,log=INLINE
/usr             /usr             namefs Jun 23 03:30 ro
/dev/fslv04      /var             jfs2   Jun 23 03:30 rw,log=INLINE
```

WPAR with root (/) partition as vxfs

Starting with the Storage Foundation 5.1SP1 release, the / (root) partition of any WPAR can be created as vxfs. Previous to this release, it was mandatory to have the / partition as JFS2. Other mount points appear as before but / can be vxfs.

```
# mkwpar -n fsqawpar -M directory=/
dev=/dev/vx/rdsk/testdg/vol2 vfs=vxfs

# startwpar -Dv fsqawpar 2>/fsqawpar_tl2

# clogin fsqawpar

# mount
```

```
node         mounted       mounted over  vfs     date         options
--------     ------------  ------------  ------ ------------ ---------------
/dev/vx/dsk/testdg/vol1      /     vxfs   Jun 23 03:30 rw,log=INLINE
/dev/fslv01    /home               jfs2   Jun 23 03:30 rw,log=INLINE
/opt          /opt                 namefs Jun 23 03:30 ro
/proc         /proc                namefs Jun 23 03:30 rw
/dev/fslv02   /tmp                 jfs2   Jun 23 03:30 rw,log=INLINE
/usr          /usr                 namefs Jun 23 03:30 ro
/dev/fslv03   /var                 jfs2   Jun 23 03:30 rw,log=INLINE
```

WPAR mobility

Live application mobility allows for planned migrations of workload from one system to another without interrupting the application. This technology can be used to perform a planned firmware installation on the server. Most workloads do not need to be aware of the WPAR relocation. WPAR mobility, also referred to as relocation, applies to both types of WPARs: application and system. The relocation of a WPAR consists of moving its executable code from one LPAR to another one while keeping the application data on the same storage devices. It is therefore mandatory that these storage devices are accessible from both the source and target LPARs hosting the WPAR. The hosting global environment hides the physical and logical device implementations from the hosted WPARs. The WPAR only works with data storage at the file system level. All files that need to be written by the application must be hosted on an NFS file system. All other files, including the AIX operating system files, can be stored in file systems local to the hosting global environment. The NFS server must provide access to both the global environment and the WPAR in order for the WPAR to work at all. In a mobility scenario, access must be provided to the WPAR and all global environments to which the WPAR might be moved.

Veritas Cluster Server agents for WPARS

There is a VCS agent for managing WPARs.

For more information, refer to the *Veritas Cluster Server Bundled Agents Reference Guide*.

Chapter 8

Configuring VCS for Workload Partitions

This chapter includes the following topics:

About VCS support for WPARs

VCS provides application management and high availability to applications that run in WPARs. VCS supports only system WPARs, application WPARs are not supported.

Overview of how VCS works with WPARs

You can use VCS to perform the following:

- Start, stop, monitor, and failover a local WPAR.
- Start, stop, monitor, and failover an application that runs in a WPAR.

Installing and configuring WPARs in VCS environments

Install and configure the WPAR. Create the service group with the standard application resource types (application, storage, networking) and the WPAR resource. The WPAR resource is how VCS represents the WPAR and its state. You then configure the service group's ContainerInfo attribute.

Configuring the ContainerInfo attribute

The service group attribute ContainerInfo specifies information about the WPAR. When you have configured and enabled the ContainerInfo attribute, you have

enabled the WPAR-aware resources in the service group to work in the WPAR environment. VCS defines the WPAR information at the level of the service group so that you do not have to define it for each resource. You can specify a per-system value for the ContainerInfo attribute.

Running VCS, its resources, and your applications

VCS and the necessary agents run in the global environment. For applications that run in a WPAR, the agents can run some of their functions (entry points) inside the WPAR. If any resource faults, VCS fails over the service group with the WPAR to another node.

The ContainerInfo attribute

The ContainerInfo attribute has the Name key, Type key, and Enabled key. The Name key defines the name of the WPAR. The Type key lets you select the type of container that you plan to use (WPAR or Zone). The Enabled key enables the WPAR-aware resources within the service group.

You can specify a per-system value for the ContainerInfo attribute. For more information, refer to the *Veritas Cluster Server Administrator's Guide*.

The ContainerOpts resource attribute

The ContainerOpts resource attribute determines the following:

- If the resource has this attribute, it is “WPAR-aware”
- Whether the WPAR-aware resource can run in the WPAR
- Whether the container information that is defined in the service group’s ContainerInfo attribute is passed to the resource. An example use of this value is to pass the agent the name of the WPAR.
  For more information, refer to the *Veritas Cluster Server Administrator's Guide*.

Note: Symantec recommends that you do not modify the value of the ContainerOpts attribute, with the exception of the Mount agent.

WPAR-aware resources

The following are the ContainerOpts attribute default values for resource types. WPAR-aware resources have predefined default values for the ContainerOpts attribute.

Table 8-1 ContainerOpts attribute default values for resource types

Resource Type	RunInContainer	PassCInfo
Application	1	0
DB2	1	0
IP	0	1
IPMultiNICB	0	1
Mount	0	0
Process	1	0
WPAR	0	1

About the Mount agent

You may need to modify the ContainerOpts values for the Mount resource in certain situations. Refer to the *Veritas Cluster Server Bundled Agents Reference Guide* for more information.

About the WPAR agent

The WPAR agent monitors WPARs, brings WPARs online, and takes them offline. For more information about the agent, see the *Veritas Cluster Server Bundled Agents Reference Guide*.

The agent creates a user account with group administrative privileges to enable communication between the global environment and the WPAR, if such an account does not exist. In secure clusters, it also renews the authentication certificate before the certificate expires.

About configuring VCS in WPARs

Configuring VCS in WPARs involves the following tasks:

- Review the prerequisites.
  See "Prerequisites for configuring VCS in WPARs" on page 71.
- Decide on the location of the WPAR root, which is either on local storage or NFS. The WPAR root is the topmost directory in a section of the file system hierarchy in which the WPAR is configured.
  See "Creating WPAR root on shared storage using NFS" on page 73.
- Install the application in the WPAR.
  See "Installing the application" on page 76.
- Create the application service group and configure its resources.
  See "Configuring the service group for the application" on page 76.

Prerequisites for configuring VCS in WPARs

- All nodes that host applications that are configured in WPARs must run the same version of the operating system.
- The WPAR root must be installed on JFS, JFS2, or NFS.
- Mounts must meet one of the following two conditions:
  - Use a loop-back file system. All mounts that the application uses must be part of the WPAR configuration and must be configured in the service group. For example, you can create a WPAR, z-ora, and define the file system containing the application's data to have the mount point as /oradata. When you create the WPAR, you can define a path in the global environment, for example /export/home/oradata, which the mount directory in the WPAR maps to. The MountPoint attribute of the Mount resource for the application is set to /export/home/oradata.
  - Use a direct mount file system. All file system mount points that the application uses that run in a WPAR must be set relative to the WPAR's root. For example, if the Oracle application uses /oradata, and you create the WPAR with the WPAR path as /z_ora, then the mount must be /z_ora/root/oradata. The MountPoint attribute of the Mount resource must be set to this path.

About using custom agents in WPARs

- If you use custom agents to monitor applications running in WPARs, make sure the agents use script-based entry points. VCS does not support running C++ entry points inside a WPAR.

- If the custom agent monitors an application that runs in a WPAR, add the resource type to the APP_TYPES environment variable. If the custom agent monitors an application running in the global environment, add the resource type to the SYS_TYPES environment variable.
- If you want the custom agent to monitor an application in the WPAR, for the custom agent type, set the following values for the ContainerOpts attribute: RunInContainer = 1 and the PassCInfo = 0.
- If you do not want the custom agent to monitor an application in the WPAR, for the custom agent type, set the following values for the ContainerOpts attribute: RunInContainer = 0 and the PassCInfo= 0.

Deciding on the WPAR root location

Each WPAR has its own section of the file system hierarchy in the WPAR root directory. Processes that run in the WPAR can access files only within the WPAR root.

You can set the WPAR root in the following two ways:

- WPAR root on local storage.
  In this configuration, you must create a WPAR on each node in the cluster.
- WPAR root on NFS.
  In this configuration, create a WPAR on the NFS storage. You need to duplicate the configuration across all the nodes in the cluster.
  When you set the WPAR root on NFS, install the WPAR from one node only. The WPAR root can fail over to the other nodes in the cluster. The system software, including the patches, must be identical on each node during the existence of the WPAR.

Creating a WPAR root on local disk

Use the following procedure to create a WPAR root on the local disk on each node in the cluster.

To create a WPAR root on local disks on each node in the cluster

1. Create the actual WPAR root directory.
2. Use the `mkwpar` command to create the WPAR.

   ```
   mkwpar -n wpar -h host -N ip_info -d wroot -o /tmp/wpar.log
   ```

   Use the following information to replace the appropriate variables:

wpar	The name of the WPAR.
hostname	The name of the system where the WPAR is created.
ip_info	Replace this variable with the information to set the virtual IP address of the system to be the IP address of the WPAR. This value also defines the device name for the NIC associated with the IP address. If you do not specify the value of the interface or netmask, the global partition's values are used. Use the following format to replace ip_info: `interface=interface netmask=netmask address=IPaddress` Example: `interface='en0' address='172.16.0.0' netmask='255.255.255.0'`
wroot	Replace this variable with the location of the WPAR root directory, for example, /wpars.

3. Repeat the command in step 2 to create the WPAR on each system in the service group's SystemList.
4. On one of the systems in the SystemList, mount the shared file system containing the application data.
5. Start the WPAR.

Creating WPAR root on shared storage using NFS

Use the following procedure to create a WPAR root on shared storage using NFS.

To create WPAR root on shared storage using NFS

1 Create a file system on NFS storage for the WPAR root. The file system that is to contain the WPAR root may be in the same file system as the file system containing the shared data.

2 Type the following `mkwpar` command to create the WPAR:

```
mkwpar -n wpar -h host -N ip_info -r -M r_fs -M v_fs -M h_fs -M
t_fs -d wroot
```

Use the following information to replace the appropriate variables:

Attribute	Description
wpar	The name of the WPAR.
host	The name of the system where the WPAR is created.
ip_info	Replace this variable with the information to set the virtual IP address of the system to be the IP address of the WPAR. This value also defines the device name for the NIC associated with the IP address. If you do not specify the value of the interface or netmask, the global partition's values are used. Use the following format to replace ip_info: `interface=interface netmask=netmask address=IPaddress` For example: `interface='en0' address='172.16.0.0' netmask='255.255.255.0'`
r_fs	Replace this variable with the information to specify the NFS volume to use for the root private file system for the WPAR. For example: `directory=/ vfs=nfs host=host123 dev=/root01`
v_fs	Replace this variable with the information to specify the NFS volume to use for the /var private file system for the WPAR. For example: `directory=/ var vfs=nfs host=host123 dev=/var01`
h_fs	Replace this variable with the information to specify the NFS volume to use for the /home private file system for the WPAR. For example: `directory=/home vfs=nfs host=host123 dev=/home01`
t_fs	Replace this variable with the information to specify the NFS volume to use for the root private file system for the WPAR. For example: `directory=/tmp vfs=nfs host=host123 dev=/tmp01`
wroot	Replace this variable with the location of the WPAR root directory, for example, /wpars.

3 Use the `lswpar` command to display information about the WPAR's properties and their values.

4 On the system where you created the WPAR, run the command:

```
mkwpar -w -o config_file_name -e wparname_just_created
```

5 On all the other systems copy the configuration file, run the command:

```
mkwpar -p -f config_file_name -n wparname_just_created
```

6 List the WPAR.

7 Start the WPAR.

8 On one system, mount the shared file system containing the application data.

9 Make sure the WPAR created from the first system is in the D state on all other systems in the service group's System List.

Installing the application

Install the application in the WPAR. Perform the following:

- If you have created WPARs on each node in the cluster, install the application identically on all nodes. If you are installing an application that supports a Veritas High Availability agent, see the installation and configuration guide for the agent.
- Install the agent. Agent packages are installed in the global environment and the currently existing WPARs. The operating system installs the agents in future WPARs when they are created.
- In the WPAR, configure all mount points used by the application.
  - If you use namefs mounts, verify the global directories are properly mounted inside the WPAR.
  - If you use a direct mount, verify the mount points used by the application have been mounted relative to the WPAR's root. For example, if a WPAR w_ora needs to use /oracle, mount the drive at /wpars/w_ora/oracle.

Configuring the service group for the application

The following diagrams illustrates different examples of resource dependencies. In one case the WPAR root is set up on local storage. In the other, WPAR root is set up on shared storage.

Figure 8-1 WPAR root on local disks (with direct mount file system)

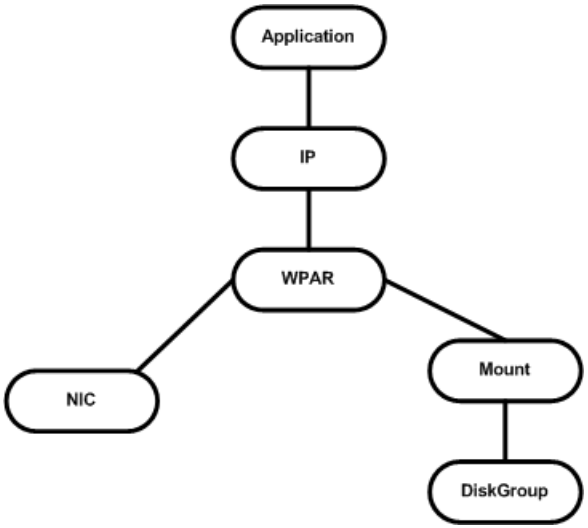


Figure 8-2 WPAR root on local disks (file system mounted from inside WPAR)

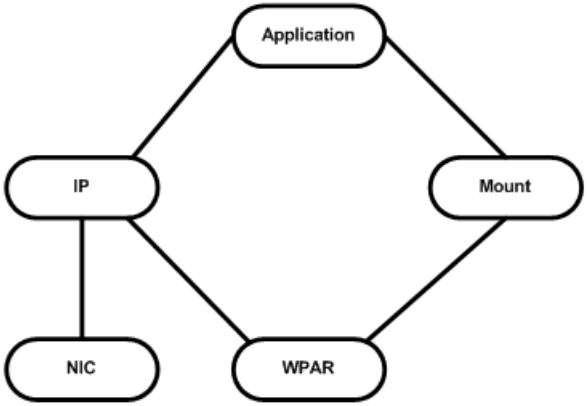
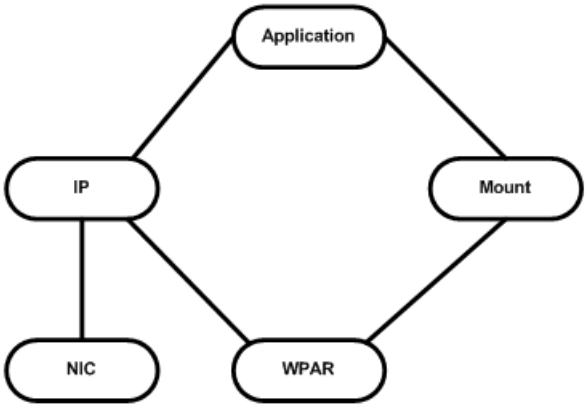


Figure 8-3 WPAR root on shared storage (with namefs file system)



Modifying the service group configuration

Perform the following procedure to modify a service group's configuration.

To modify the service group configuration

1 Run the hawparsetup script to set up the WPAR configuration.

```
# hawparsetup servicegroup_name WPARres_name WPAR_name
passwordsystems
```

servicegroup_name	Name of the application service group.
WPARres_name	Name of the resource configured to monitor the WPAR.
WPAR_name	Name of the WPAR.
password	Password to be assigned to VCS or Security (Symantec Product Authentication Service) user created by the command.
systems	List of systems on which the service group will be configured. Use this option only when creating the service group.

The script adds a resource of type WPAR to the application service group. It also creates a user account with group administrative privileges to enable inter-WPAR communication.

If the application service group does not exist, the script creates a service group with a resource of type WPAR.

2 Modify the resource dependencies to reflect your WPAR configuration. See the resource dependency diagrams for more information.

3 Save the service group configuration and bring the service group online.

Verifying the WPAR configuration

Run the `hawparverify` command to verify the WPAR configuration. The command verifies the following requirements:

- The systems hosting the service group have the required operating system to run WPARs.
- The service group does not have more than one resource of type WPAR.
- The dependencies of the WPAR resource are correct.

To verify the WPAR configuration

1. If you use custom agents make sure the resource type is added to the APP_TYPES or SYS_TYPES environment variable.

   See "About using custom agents in WPARs" on page 71.

2. Run the `hawparverify` command to verify the WPAR configuration.

   # `hawparverify` *`servicegroup_name`*

Maintenance tasks

Perform the following maintenance tasks when you use WPARs:

- Whenever you make a change that affects the WPAR configuration, you must run the `hawparsetup` command to reconfigure the WPARs in VCS.
  See "Configuring the service group for the application" on page 76.
- Make sure that the WPAR configuration files are consistent on all the nodes at all times.
- When you add a patch or upgrade the operating system on one node, make sure to upgrade the software on all nodes.
- Make sure that the application configuration is identical on all nodes. If you update the application configuration on one node, apply the same updates to all nodes.

Troubleshooting information

Symptom	Recommended Action
VCS HA commands do not work.	Verify that the VCS packages are installed. Run the `hawparsetup` script to set up the WPAR configuration. Run the`hawparverify` command to verify the configuration. Run the `halogin` command from the WPAR. For more information, refer to the *Veritas Cluster Server Administrator's Guide*. Verify your VCS credentials. Make sure the password is not changed. Verify the VxSS certificate is not expired.

Symptom	Recommended Action
Resource does not come online in the WPAR.	Verify VCS and the agent packages are installed correctly. Verify the application is installed in the WPAR. Verify the configuration definition of the agent. Make sure to define the ContainerType and ContainerName attributes.

About VCS support for Live Partition Mobility

You can use Live Partition Mobility to perform a stateful migration of an LPAR in a VCS environment. During this period, you may see notifications if the migrating node is unable to heartbeat with its peers within LLT's default peer inactive timeout. To avoid false failovers, determine how long the migrating node is unresponsive in your environment. If that time is less than the default LLT peer inactive timeout, VCS operates normally. If not, increase the peer inactive timeout to an appropriate value on all the nodes in the cluster before beginning the migration. Reset the value back to the default after the migration is complete.

For more information, refer to the *Veritas Cluster Server Administrator's Guide.*

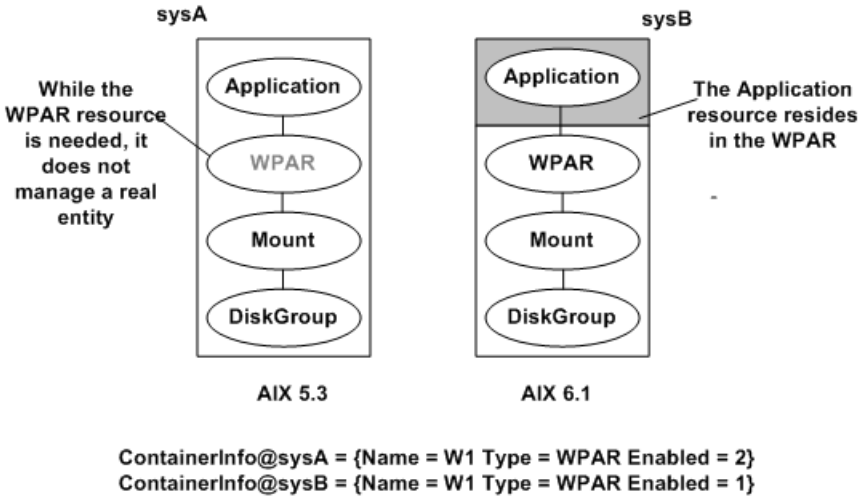
About configuring failovers among physical and virtual servers

You can configure VCS to fail over from a physical system to a virtual system and vice versa. A physical to virtual failover gives an N + N architecture in an N + 1 environment. For example, several physical servers with applications can fail over to containers on another physical server.

Configuring for failovers—a typical setup

In this configuration, you have two physical nodes. One node runs AIX 5.3 (sysA) and another node that runs AIX 6.1 (sysB). The node that runs AIX 6.1 has WPARs configured.

Figure 8-4 An application service group that can fail over onto a WPAR



In the `main.cf` configuration file, define the container name, type of container, and whether it is enabled or not. The following is an example of the ContainerInfo lines in the `main.cf` file:

```
ContainerInfo@sysA = {Name = W1, Type = WPAR, Enabled = 2}

ContainerInfo@sysB = {Name = W1, Type = WPAR, Enabled = 1}
```

On sysA, you set the value of Enabled to 2 to ignore WPARs so that the application runs on the physical system. When sysA fails over to sysB, the application runs inside the WPAR after the failover because Enabled is set to 1 on sysB. The application can likewise fail over to sysA from sysB.

Chapter 9

Data migration from Physical to Virtual Clients with NPIV

This chapter includes the following topics:

About migration from Physical to VIO environment

Symantec has qualified migration of storage that is used by Storage Foundation from the physical environment to IBM VIO environment.

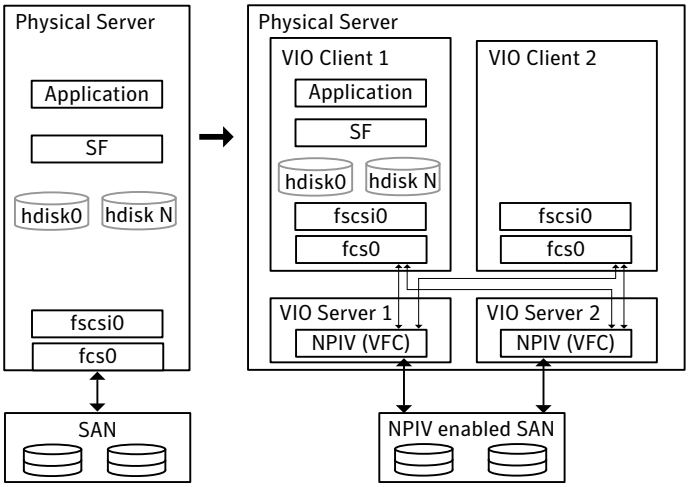
Storage Foundation provides the PDC (Portable Data Container) feature, which enables migrating storage from other platforms (Solaris, AIX, HP or Linux) to AIX VIO environment. You can also use PDC feature to migrate the storage consumed by a AIX physical server to a AIX VIO environment. NPIV helps you migrate the applications along with storage from a AIX physical environment to AIX VIO environment and vice-versa.

When storage is consumed by SF, Veritas Volume Manager (VxVM) initializes the storage LUNs as CDS (Cross-platform Data Sharing) type disks by default. A CDS disk group can be imported in a VIO client which has access to LUN's that are mapped through VFC Adapter on the client.

As part of the migration qualification, an application's storage is migrated from physical server to VIO environment (VIO client 1) which has NPIV capable FC adapter connected to it. This allows the application to access the storage in VIO

client 1. With NPIV capable FC adapter at VIOS, the devices presented to the VIO client would appear as regular AIX hdisk devices. Figure 9-1 shows this migration.

Figure 9-1 SF migration from a physical environment to AIX VIO environment



Migration is an offline task.

Storage Foundation requirement

Both the source and the target must have the same version of Storage Foundation. The version must be at least 5.0 MP3 RP1.

Migrating from Physical to VIO environment

Migration is an offline task. The migration procedure involves stopping the application, unmounting the file systems and deporting the disk group on the physical server. Prior to being deported, you can take a space optimized snapshot, to facilitate fail-back.

Verify that the devices are visible on VIO client and the VFC adapter mapping between VIOS and VIO client is set up correctly. Refer to the IBM documentation for details. After all the required devices are accessible in VIO client 1, import the disk group in the client, mount the file system, and then start the application on the VIO client 1.

Refer to *IBM* documentation on how to configure the VFC adapter mappings between the VIO partition and the Client Partition.

After all the required devices are visible on the target Client mLPAR, the application disk group can be imported, and the file systems mounted.

Chapter 10

Boot device management

This chapter includes the following topics:

Using DMP to provide multi-pathing for the root volume group (rootvg)

In many cases, the use of MPIO for the rootvg creates a situation with dual multi-pathing tools. To simplify system administration and system reliability, use DMP to provide multi-pathing for the rootvg.

This release supports using DMP for the rootvg on vSCSI, NPIV, and physical HBAs. Starting with release 5.1, DMP provides support for alternate root disks and root disks with multiple volumes.

To use DMP on the rootvg, DMP requires a vendor-specific ODM predefined fileset. Symantec includes the predefined filesets for vSCSI devices in the Veritas product distribution. For other devices, obtain and install the ODM predefined fileset from the storage vendor. For example, for the IBM DS array, install the `devices.fcp.disk.ibm.rte` fileset.

http://www-1.ibm.com/support/docview.wss?rs=540&context=ST52G7&dc=D400&q1=host+script&uid=ssg1S4000199&loc=en_US&cs=utf-8&lang=en

In this release, with the help of OS Native stack support feature, rootability is achieved by using the `vxdmpadm` command. In previous releases, rootability was achieved through the use of the `vxdmproot install` command.

To get help about rootability

- Run the following command:

```
# vxdmpadm help native
Manage DMP support for AIX boot volume group(rootvg)
Usage:
vxdmpadm native { enable | disable } vgname=rootvg
vxdmpadm native list [ vgname=rootvg ]
where,
enable Enable DMP support for AIX boot volume group(rootvg)
list List boot paths on which DMP support is enabled
disable Disable DMP support for AIX boot volume group(rootvg)
```

To enable rootability

1. Run the following command:

```
# vxdmpadm native enable vgname=rootvg
```

2. Reboot the system to enable DMP support for LVM bootablity.

To disable rootability

1. Run the following command:

```
# vxdmpadm native disable vgname=rootvg
```

2. Reboot the system to disable DMP support for LVM bootability.

To monitor rootability

1. Run the following command:
2.
```
# vxdmpadm native list
PATH          DMPNODENAME
==========================
hdisk64       ams_wms0_302
hdisk63       ams_wms0_302
```

Boot device on NPIV presented devices, NPIV for data volumes

Storage Foundation supports the use of boot from NPIV presented devices with the use of DMP for the rootvg, within the requirements outlined in the vendor support matrix.

Hardware and software requirements

- Any Power 6 based computer
- SAN Switch & FC Adapters should be NPIV capable.
- At least one 8 GB PCI Express Dual Port FC Adapter in VIOS.
- VIOC Minimum OS-level : (i) AIX 6.1 TL2 or later (ii) AIX 5.3 TL9 or later
- VIO Server Version 2.1 with Fix Pack 20.1 or later
- HMC 7.3.4

Boot Device Management

All the LUNs presented through NPIV for a client LPAR has the characteristics of a dedicated HBA. There for the procedure for using DMP on rootvg devices from NPIV presented devices is similar to using DMP on rootvg devices from physical HBA. Use of DMP on rootvg is supported from 5.0 and 5.0 MP3 through `vxdmproot enable` command. From 5.1 the `vxdmproot` command is deprecated and the same functionality is achieved through `vxdmpadm native enable vgname=rootvg`.

The following steps are to be followed for using DMP on rootvg:

To use DMP on rootvg

1. Storage vendor supplied ODM pre-defines should be installed to disable MPIO.
2. Run the `vxdmpadm native enable vgname=rootvg` command.

NPIV for Data volumes

The behavior of Data volumes presented through NPIV is similar to that of physical HBA. No special handling is required for these volumes. All SCSI device inquiry operations work and SCSI-3 persistent reservation functionality is also supported, enabling the use of SCSI-3 I/O Fencing if the underlying storage supports.

Glossary

Active Memory™ Sharing - Statement of Direction	Provides the ability to pool memory across micro-partitions which can be dynamically allocated based on partition's workload demands to improves memory utilization.
Dynamic Logical Partition (DLPAR)	A virtual server with the ability to add or remove full processors, network, or storage adapters while the server remains online.
Hardware Management Console (HMC)	Dedicated hardware/software to configure and administer a partition capable POWER server.
Integrated Virtualization Manager	Management console which runs in the VIO for partition management of entry level systems.
Live Partition Mobility	Provides the ability to migrate running AIX and Linux partitions across physical servers.
Lx86	Supports x86 Linux applications running on POWER.
Logical Partition (LPAR)	A virtual server running its own operating system instance with dedicated processors and I/O adapters.
Micro-partition	A virtual server with shared processor pools with support for up to10 micro-partitions per processor core. Depending upon the Power server, you can run up to 254 independent micro-partitions within a single physical Power server. Processor resources can be assigned at a granularity of 1/100th of a core. Also known as shared processor partition.
Multiple Shared Processor Pools	Shared and capped processor resources for a group of micro-partitions.
N_Port ID Virtualization (NPIV)	Virtual HBA's which enable multiple LPARs/micro-partitions to access SAN devices thru shared HBA's providing direct Fiber Channel connections from client partitions to storage. Fiber Channel Host Bus Adapters (HBAs) are owned by VIO Server partition.
POWER Hypervisor	responsible for dispatching the logical partition workload across the shared physical processors. The POWER Hypervisor also enforces partition security, and provides inter-partition communication that enables the Virtual I/O Server's virtual SCSI and virtual Ethernet function.
Shared Ethernet Adapter	Enables network traffic outside the physical server by routing it through a software-based layer 2 switch running in the VIO Server.

Virtual I/O Server (VIO) A dedicated LPAR which supports the I/O needs of client partitions (AIX and Linux) without the need to dedicate separate I/O slots for network connections and storage devices for each client partition.

Virtual Ethernet In-memory network connections between partitions by POWER Hypervisor that reduces or eliminates the need for separate physical Ethernet Adapters in each LPAR.

Virtual SCSI Virtual Disks provided by the VIO server to reduce the need for dedicated physical disk resources for client partitions. HBA's are contained in the VIO server. vDisks can be full LUNs or logical volumes. Dynamic LPARs or micro-partitions can also use dedicated HBAs

WPAR Application Partition An application Partition is a light weight partition for running individual applications in. Can only run application processes, not system daemons such as `inetd` or `cron`. Temporary object which is removed when app is completed.

WPAR System Partition A WPAR System partition has a private copy of many of the AIX OS parameters. If desired, it can have its own dedicated, completely writable file systems. Most OS daemons can run, and each System Partition has its own user privilege space. By default, a System Partition has no access to physical devices.

WPAR Manager Allows an administrator to create, clone, and remove WPAR definitions, or start and stop WPARs. Enables Live Application Mobility which allows relocation of WPARs from one server to another without restarting the application. The WPAR Manager Includes a policy engine to automate relocation of WPARs between systems based on system load and other metrics.