Web Exclusive



# VIOS 101: I/O

August 2015 | by Jaqui Lynch

In Part 1, I wrote an introduction to virtualization and then discussed CPU virtualization with PowerVM. In Part 2, I addressed memory technologies and in Part 3, we looked at network operations. In this final article, we'll review options for virtualizing I/O.

## What Are Our I/O Options?

Virtualization of I/O for LPARs comes in many flavors. Options include virtual SCSI (vSCSI), virtual Fibre Channel (VFC), virtual Optical (vtOpt), virtual Tape (vtTape) and shared storage pools (SSPs). VFC uses N-Port ID Virtualization (NPIV), which allows multiple Fibre Channel node port (N_Port) IDs to share a single physical N_Port, thus allowing multiple LPARs to share a single physical Fibre adapter port.

To provide a boot disk to an LPAR, it's necessary to take up a slot for a card that connected the disk and that slot gets dedicated to the LPAR. As servers get larger and faster, more LPARs are being consolidated onto them and this is causing a significant increase in the amount of disk and the number of slots being consumed, just to boot the LPAR, never mind the actual data disks. vSCSI and NPIV (VFCs) provide the capability to have a VIO server own the adapters and the disks, and then that VIO server can carve those disks up and provide chunks of disks (or whole disks) to LPARs such that the client LPAR thinks it has a full disk that it can use for booting or data. As an example, most LPARs rootvgs are between 30 and 45 GB depending on how clean they keep their rootvg. The smallest disk now is around 146 GB. With a VIO server this 146 GB disk can easily be carved into three logical volumes, each of which could be a boot volume for a different LPAR. Even using two VIO servers for redundancy this is still a significant reduction in disks, PCI cards and I/O drawers.

Also note that a physical adapter can support both NPIV and vSCSI concurrently. Some people like to use vSCSI for boot disks and NPIV for data. They do this because of the requirement for MPIO (multipath I/O) drivers to be in the client LPAR for NPIV, which can make things tricky when upgrading the operating system and the MPIO drivers. Any of the three options (NPIV only, vSCSI only or both) works just fine as long as you pay attention to upgrade and other requirements.

vSCSI doesn't support load balancing across virtual adapters in a client LPAR. With VFCs, device drivers such as SDD, SDDPCM or Atape must be installed in the client partition for the disk devices or tape devices. SDD or SDDPCM allow load balancing across virtual adapters. However, upgrading of these drivers requires special attention when you're using SAN devices as boot disks for the operating system. MPIO for VFC devices in the AIX client partition doesn't require any specific configuration and supports round robin, load balancing and failover modes. MPIO for vSCSI devices in the AIX client partition only supports failover mode. For any given vSCSI disk, a client partition will use a primary path to one VIO server and fail over to the secondary path to use the other VIO server.

## Virtual SCSI

vSCSI allows the VIO server to carve up the storage on a Fibre adapter and allocate it to various client LPARs. The vSCSI protocol supports connections over Fibre Channel, parallel SCSI and SCSI RAID devices. It also provides for optical devices such as DVD drives. There is a method to virtualize tape as

well. It should be noted that vSCSI requires more overhead in the VIO server than NPIV does – this is because vSCSI requires the VIO server to handle the I/O traffic that goes through the I/O stack in the VIO server. vSCSI supports all of the mandatory commands in the SCSI protocol but not all optional commands are supported. So it's important to use man pages or the commands reference to check the syntax for commands to implement vSCSI.

When vSCSI and virtual Ethernet are both being used on the same VIO server it should be noted that Virtual Ethernet, having non-persistent traffic, runs at a higher priority than the vSCSI. To make sure networking traffic doesn't starve vSCSI of CPU cycles, it's important that threading is turned on. Threading provides the best balance of mixed throughput and has been the default since PowerVM v1.2.

With vSCSI, you have a couple of options for allocating out the disk. One option is to take a provided LUN (hdisk) and carve it up into LVs (logical volumes). The LVs can then be allocated to individual LPARs. While this works great, it's not permitted if you want to use LPM (Live Partition Mobility). Storage used can be internal or on the SAN but only SAN-provided disk is supported for LPM or SSPs. For the most part, the preferred option is the one where the whole LUN or hdisk comes from the SAN and is allocated to the LPAR. With vSCSI the MPIO drivers are installed in the VIO servers as are the device drivers for the storage. This can result in some savings if the vendor charges for its MPIO drivers. In a dual-VIO environment, the same SAN disk would be allocated from each VIO server and AIX-native MPIO would be used in the client to provide multipathing. Mappings can be checked as follows:

```
lsmap -all
```

vSCSI is required for certain functions such as vtOpt and SSPs. VFC doesn't support virtualization capabilities that are based on the SSP, such as thin provisioning.

### Virtual Fibre Channel and NPIV

As mentioned, NPIV allows multiple LPARs to share a single Fibre channel adapter port, which allows for consolidation of Fibre adapters and frees up slots in the servers and I/O drawers. With vSCSI, the VIO server owns the adapters and sees all storage. Storage then gets mapped at the VIO server to the various client LPARs, which results in the need to keep spreadsheets to track allocations. This leaves the potential for mapping errors. With NPIV, the VIO server still owns the Fibre adapters, but the virtual adapters are owned by the client LPAR. The client LPAR sees only its own storage and the VIO server doesn't see that storage. This makes zoning and allocating storage easier and cleaner. Another difference is that with NPIV the MPIO drivers are installed in the client LPAR not the VIO server so this can lead to increased costs if the vendor charges for those drivers.

For VFC (NPIV), the VIO server acts as a Fibre channel pass-through whereas vSCSI acts as a SCSI emulator. Two unique virtual worldwide port names (WWPNs) starting with the letter c are generated by the HMC (hardware management console) or IVM (integrated virtualization manager) for the VFC client adapter. Once the client LPAR is activated those WWPNs log in to the SAN like any other WWPNs from a physical port so that disk or tape storage target devices can be assigned to them as if they were physical Fibre channel ports. A VFC client adapter is a virtual device that provides VIO client partitions with a Fibre Channel connection to a SAN through the VIO server partition.

NPIV implementation has some specific requirements. While vSCSI can work on any of the supported adapters, NPIV requires adapters that support NPIV. Currently, these include the 8 Gigabit dual and four port Fibre adapters, the 16 Gigabit Fibre adapter and some of the new 10 Gigabit FCoE adapters. NPIV also requires that the closest switch connected to the server provides for NPIV support. The lsnports command can be used to check for this.

When using VFCs, you need to define one VFC client adapter per Fibre adapter port that you want to connect to the LPAR. These are then mapped to the LPAR from the VIO server. Mappings can be checked with:

```
lsmap –all –npiv
```

There is a maximum of 64 active VFC client adapters per physical port and a maximum of 32,000 unique WWPN pairs per system. When using NPIV, it's better to reuse partition profiles rather than deleting them as deleting them makes those unique WWPNs unavailable for future use.

### Implementation and Other Considerations for vSCSI and NPIV

A VFC server adapter needs to be created by the HMC for the VIO server partition profile that connects to a VFC client adapter created in the client LPAR. In the case of IVM, the server adapter is automatically created when the client adapter is created. Normally the server adapter is created first and then the client adapter.

The VIO server LPAR provides the connection between the VFC server adapters and the physical Fibre Channel adapters assigned to the VIO server partition on the managed system. Mapping is done using the vfcmap command.

When a client LPAR is created. it's assigned a vhost number – vSCSI mappings are assigned by vhost. When a VFC adapter is created, it's assigned a vfchost number and NPIV mappings are assigned by vfchost number. To assign storage, commands would be similar to the following:

Setup NPIV mappings:

```
vfcmap –vadapter vfchost0 –fcp fcs0
vfcmap –vadapter vfchost1 –fcp fcs1
```

These map two physical Fibre adapter ports to two VFC adapters.

Commands to show the mappings and information on the adapters:

```
lsmap –npiv –all
lsmap –vadapter vfchost0 –npiv
lsmap –vadapter vfchost1 –npiv
lsdev –virtual
lsnports
lsdev –slots
lscfg –vpl vfchost0
lscfg –vpl vfchost1
```

Setup vSCSI mappings:

```
mkvdev –vdev hdisk5 –vadapter vhost0
```

This maps hdisk5 on the VIO server to vhost0.

Commands to show the mappings and information on the adapters:

```
lsmap –all
lsmap –vadapter vhost0
lsdev –virtual
lscfg –vpl vhost0
lsattr –El hdisk5
```

With vSCSI and NPIV, it's important to ensure that all disks are set with reserve_policy=no_reserve. With vSCSI, you should also check queue_depth on the VIO server for each hdisk and on the client as well.

The client will most likely default to the SCSI value of 3 and you may need to increase this. Don't make it higher than whatever it is on the VIO server. For NPIV, queue_depth is only set on the hdisks on the client LPAR.

Fibre adapter tuning settings, specifically num_cmd_elems and max_xfer_size, are set on the Fibre adapters using chdev against the fcs. For vSCSI, these are only set on the VIO server. For NPIV, these are set on the VIO server and are also on the client LPAR. In recent releases of AIX, the client LPAR has a maximum setting of 256 (default is 200) for num_cmd_elems for NPIV clients. The VIO server settings must be at least as high as those for the client LPARs and must be set and activated (normally a VIO server reboot) prior to changing the client LPARs. It's not uncommon to see num_cmd_elems set to 1024 or 2048 on a VIO server instead of the default 200. Typically, the command to change settings on the Fibre adapters is similar to:

```
chdev -l fcs0 -a max_xfer_size=0x200000 -a num_cmd_elems=1024 -P
```

### Virtual Optical

VtOpt, also known as file backed optical (FBO), was introduced in PowerVM v1.5 and allows the VIO server to take a DVD that's assigned to it and to virtualize it for shared use by the client LPARs. Additionally vtOpt can be used to provide FBO. This allows for loading ISO images of DVDs into a repository on the VIO server and then sharing those images out to client LPARs. The client LPAR sees those images as if they were a CD/DVD image.

While only one virtual I/O client partition can have access to the drive at a time, the advantage of a vtOpt device is that you don't have to move the parent SCSI adapter between VIO clients. In many cases, this wouldn't be possible anyway as the SCSI adapter often controls the internal disk drives on which the VIO server is installed. The virtual drive can't be shared with another VIO server as client SCSI adapters (required for vtOpt) can't be created in a VIO server.

To use FBO, it's best to add a separate disk to the VIO server and put it into its own volume group. This ensures that any mksysb image of the VIO server is not huge. If that isn't an option, then ensure you use the –nomedialib flag on your backups if you want to be able to restore quickly.

Once the volume group is created you can use mkrep to create the repository:

```
mkrep -sp fbovg -size 500g
```

This creates a 500 GB FBO library in the volume group fbovg. lssp will list the pool.

FBO requires that a vSCSI connection be available to the vhost. Once that's there, we can create an FBO device as follows:

```
mkvdev –fbo –vadapter vhost0
```

Since this is the first FBO device I created, the command creates a device called vtopt0.

Images are loaded into the repository using the mkvopt command and are normally loaded from an ISO image of the DVD.

```
mkvopt -name aix71base1 -file /software/aix71tl04sp4-base1.iso
```

This loads the AIX v7.1 tl04 sp4 disk 1 image into the repository and names it aix71base1. I had previously ripped that disk to an ISO image.

I can make that image available to vhost0 (which is vtopt0) as follows:

```
loadopt -disk aix71base1 -vtd vtopt0
```

On the client LPAR, I can now see that image if I look at /dev/cd0.

FBO is a very useful way to provide CD/DVD images to client LPARs without having to move DVD drives around.

### Virtual Tape

A tape device attached to a VIO server can be virtualized and assigned to VIO clients. This is done by assigning the physical tape drive to the VIO server partition and then creating a vSCSI server adapter using the HMC to which any partition can connect. A vtTape is defined on the VIO using:

```
mkvdev -vdev rmt0 –vadapter vhost0
```

This creates vttape0 as a virtual version of rmt0 and assigns it to vhost0. Most new servers don't have internal tape drives so vtTape is not as commonly in use as FBO.

### Shared Storage Pools

SSPs became available at PowerVM 2.2.0.11 SP1 and refer to a pool of SAN storage devices that can be shared among multiple VIO servers and their client LPARs. It's based on a cluster of VIO servers that are cluster nodes. The cluster includes a distributed data object repository and a global namespace for keeping track of what's happening. The SSP concept takes advantage of the Cluster Aware AIX (CAA) feature in AIX along with RSCT to form a cluster of VIO servers. It's a server-based approach to storage virtualization that simplifies the aggregation of large numbers of disks across multiple VIO servers, simplifies the administration of that storage and helps improve storage utilization by using thin provisioning. Thin provisioning means that the device is not fully backed by physical storage if the data block is not in actual use – physical storage gets assigned when it's actually required, not at definition time. To protect against over-allocating space, the VIO server posts a warning message when the pool has less than 75 percent free space. Features provided by SSPs include thick provisioning, thin provisioning and snapshot features.

When using SSPs, storage is provided by the VIO servers through logical units that get assigned to client LPARs. A logical unit is a file-backed storage device that exists in the cluster filesystem in the SSP and it can't be resized after creation. It appears as a vSCSI device in the client LPAR so vSCSI is a prereq for SSPs. VFC doesn't support virtualization capabilities based on the SSP, such as thin provisioning. Additionally, all VIO servers in the cluster must have a network connection to each other. All cluster nodes can see all the disks so disks must therefore be zoned to all cluster nodes that are part of the SSPs. The poold daemon handles group services and the vio_daemon monitors the health of the cluster nodes and the pool as well as pool capacity.

### Many Ways

As you can see, you have multiple ways to support storage with the VIO servers. Apart from dedicated adapters, we also have the option of vSCSI, VFC/NPIV and SSPs for our SAN-provided disk. And we have the capability to virtualize DVD drives as well as tape drives. Finally, we have the option to provide ISO images to an LPAR by virtualizing them using FBO so they appear to the client LPAR as if they were a DVD drive. The options are very flexible and most of them can be used at the same time if so desired.

# References

< Return to main article



Print Email

PowerVM Enhancements in 2013
http://publib-b.boulder.ibm.com/abstracts/sg248198.html?Open

PowerVM Virtualization Introduction and Configuration
http://publib-b.boulder.ibm.com/abstracts/sg247940.html?Open

PowerVM Virtualization Managing and Monitoring
http://publib-b.boulder.ibm.com/abstracts/sg247590.html?Open

PowerVM Best Practices
http://publib-b.boulder.ibm.com/abstracts/sg248062.html?Open

VIOS Care and Feeding Replay
http://www.circle4.com/movies/

< Return to main article