

Tips for implementing NPIV on IBM Power Systems

Virtual Fibre Channel with Virtual I/O and AIX 6.1

Skill Level: Introductory

[Chris Gibson](#)
AIX Specialist
Southern Cross Computer Systems

11 Oct 2011

Chris Gibson shares some tips for implementing NPIV in an AIX and Virtual I/O Server environment on IBM POWER7 systems.

Overview

In this article, I will share with you my experience in implementing NPIV on IBM Power Systems with AIX and the Virtual I/O Server (VIOS). There are several publications that already discuss the steps on how to configure NPIV using a VIOS, and I have provided links to some of these in the [Resources](#) section. Therefore, I will not step through the process of creating virtual Fibre Channel (FC) adapters or preparing your environment so that it is NPIV and virtual FC ready. I assume you already know about this and will ensure you have everything you need. Rather, I will impart information that I found interesting and perhaps undocumented during my own real-life experience of deploying this technology. Ultimately this system was to provide an infrastructure platform to host SAP applications running against a DB2 database.

NPIV (N_Port ID Virtualization) is an industry standard that allows a single physical Fibre Channel port to be shared among multiple systems. Using this technology you can connect multiple systems (in my case AIX LPARs) to one physical port of a physical fibre channel adapter. Each system (LPAR) has its own unique worldwide port name (WWPN) associated with its own virtual FC adapter. This means you can

connect each LPAR to physical storage on a SAN natively.

This is advantageous for several reasons. First, you can save money. Having the ability to share a single fibre channel adapter among multiple LPARs could save you the cost of purchasing more adapters than you really need.

Another reason to use NPIV is the reduction in VIOS administration overhead. Unlike virtual SCSI (VSCSI), there is no need to assign the SAN disks to the VIOS first and then map them to the Virtual I/O client (VIOC) LPARs. Instead, the storage is zoned directly to the WWPNs of the virtual FC adapters on the clients. It also eliminates the need to keep your documentation up to date every time you map a new disk to an LPAR/VIOS or un-map a disk on the VIO server.

I/O performance is another reason you may choose NPIV over VSCSI. With NPIV all paths to a disk can be active with MPIO, thus increasing the overall bandwidth and availability to your SAN storage. The I/O load can be load-balanced across more than one VIO server at a time. There is no longer any need to modify a client's VSCSI hdisk path priority to send I/O to an alternate VIO server, as all I/O can be served by all the VIO servers if you wish.

One more reason is the use of disk "copy service" functions. Most modern storage devices provide customers with the capability to "flash copy" or "snap shot" their SAN LUNs for all sorts of purposes, like cloning of systems, taking backups, and so on. It can be a challenge to implement these types of functions when using VSCSI. It is possible, but automation of the processes can be tricky. Some products provide tools that can be run from the host level rather than on the storage subsystem. For this to work effectively, the client LPARs often need to "see" the disk as a native device. For example, it may be necessary for an AIX system to detect that its disk is a native NetApp disk for the NetApp "snapshot" tools to work. If it cannot find a native NetApp device, and instead finds only a VSCSI disk, and it is unable to communicate with the NetApp system directly, then the tool may fail to function or be supported.

The biggest disadvantage (that I can see) to using NPIV is the fact that you must install any necessary MPIO device drivers and/or host attachment kits on any and all of the client LPARs. This means that if you have 100 AIX LPARs that all use NPIV and connect to IBM DS8300 disk, you must install and maintain SDDPCM on all 100 LPARs. In contrast, when you implement VSCSI, the VIOS is the only place that you must install and maintain SDDPCM. And there's bound to be fewer VIOS than there are clients! There are commonly only two to four VIO servers on a given Power system.

Generally speaking, I'd recommend NPIV at most large enterprise sites since it is far more flexible, manageable, and scalable. However, there's still a place for VSCSI, even in the larger sites. In some cases, it may be better to use VSCSI for the rootvg disk(s) and use NPIV for all non-rootvg (data) volume groups. For example, if you

boot from SAN using NPIV (rootvg resides on SAN disk) and you had to install MPIO device drivers to support the storage. It can often be difficult to update MPIO software when it is still in use, which in the case of SAN boot is all the time. There are procedures and methods to work around this, but if you can avoid it, then you should consider it!

For example, if you were a customer that had a large number of AIX LPARs that were all going to boot from HDS SAN storage, then I'd suggest that you use VSCSI for the rootvg disks. This means that HDLM (*Hitachi Dynamic Link Manager*, HDS MPIO) software would need to be installed on the VIOS, the HDS LUNs for rootvg would be assigned to and mapped from the VIOS. All other LUNs for data (for databases or application files/code) would reside on storage presented via NPIV and virtual FC adapters. HDLM would also be installed on the LPARs but only for non-rootvg disks. Implementing it this way means that when it comes time to update the HDLM software on the AIX LPARs, you would not need to worry about moving rootvg to non-HDS storage so that you can update the software. Food for thought!

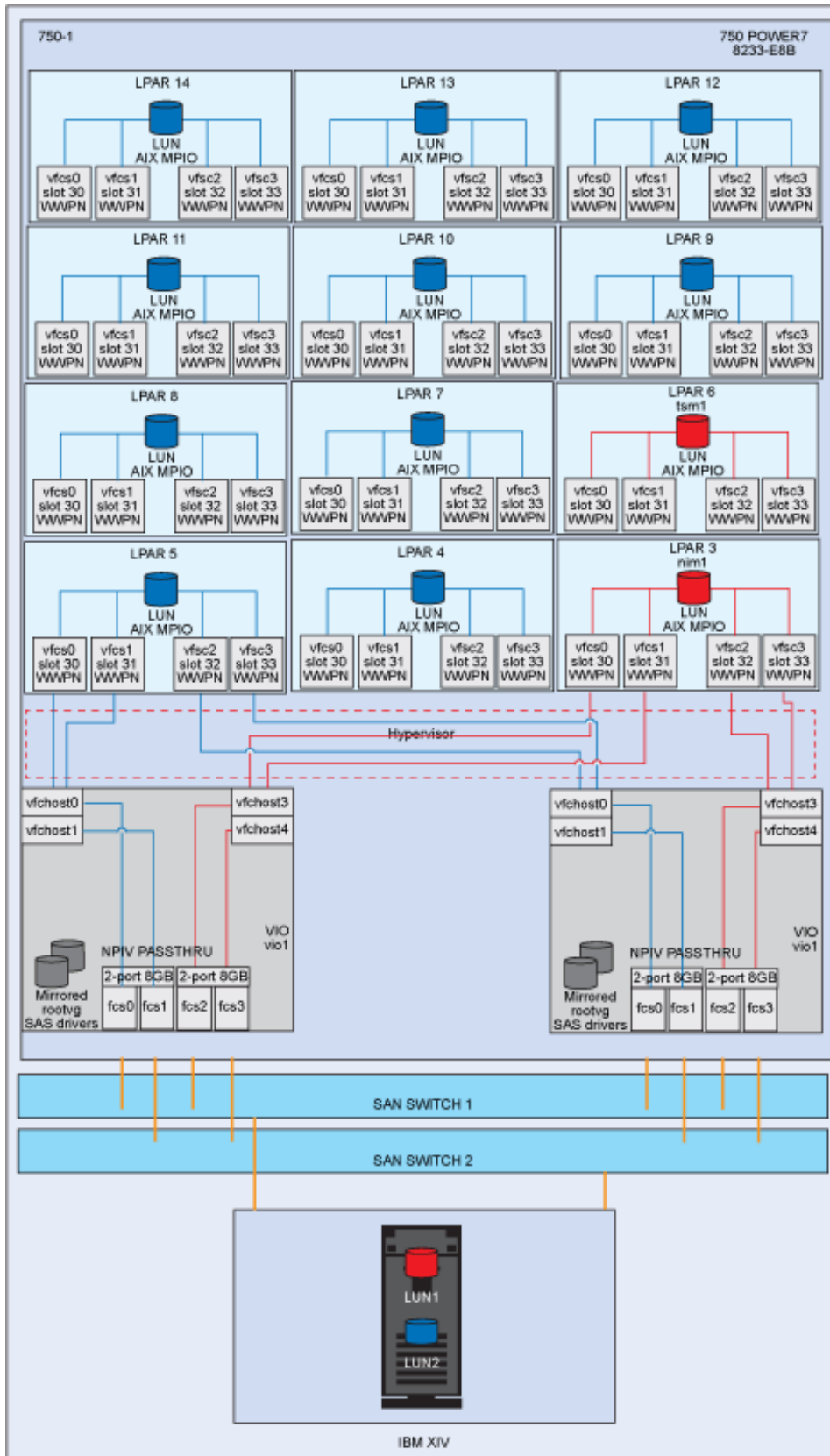
Environment

The environment I will describe for my NPIV implementation consists of a POWER7 750 and IBM XIV storage. The client LPARs are all running AIX 6.1 TL6 SP3. The VIO servers are running version 2.2.0.10 Fix Pack 24 Service Pack 1 (2.2.0.10-FP-24-SP-01). The 750 is configured with six 8GB fibre channel adapters (feature code 5735). Each 8GB FC adapter has 2 ports. The VIO servers were assigned 3 FC adapters each. The first two adapters in each VIOS would be used for disk and the last FC adapter in each VIOS would be for tape connectivity.

NPIV and virtual FC for disk

I made the conscious decision during the planning stage to provide each production LPAR with four virtual FC adapters. The first two virtual FC adapters would be mapped to the first two physical FC ports on the first VIOS and the last two virtual FC adapters would be mapped to first two physical FC ports on the second VIOS. As shown in the following diagram below.

Figure 1: Virtual FC connectivity to SAN and Storage



(View a [larger version](#) of Figure 1.)

I also decided to isolate *other* disk traffic (for example, non-critical production traffic) over different physical FC adapters/ports. In the previous diagram, the blue lines/LUNs indicate production traffic. This traffic is mapped from the virtual adapters, fcs0 and fcs1 in an LPAR, to the physical ports on the first FC adapters in vio1: fcs0 and fcs1. The virtual FC adapters, fcs2 and fcs3 in an LPAR, map to the physical ports on the first FC adapter in vio2: fcs0 and fcs1.

The red lines indicate all non-critical disk traffic. For example, the NIM and Tivoli Storage Manager LPARs use different FC adapters in each VIOS than the production LPARs. The virtual FC adapters, fcs0 and fcs1, map to the physical ports on the second FC adapter, fcs2 and fcs3 in vio1. The virtual FC adapters, fcs2 and fcs3, map to the physical ports on the second FC adapter, fcs2 and fcs3 in vio2.

An example of the **vfcmmap** commands that we used to create this mapping on the VIO servers are shown here:

For production systems (e.g. LPAR4):

1. Map LPAR4 vfchost0 adapter to physical FC adapter fcs0 on **vio1**.

```
$ vfcmmap -vadpater vfchost0 -fcp fcs0
```

2. Map LPAR4 vfchost1 adapter to physical FC adapter fcs1 on **vio1**.

```
$ vfcmmap -vadapter vfchost1 - fcp fcs1
```

3. Map LPAR4 vfchost0 adapter to physical FC adapter fcs0 on **vio2**.

```
$ vfcmmap -vadapter vfchost0 - fcp fcs0
```

4. Map LPAR4 vfchost1 adapter to physical FC adapter fcs1 on **vio2**.

```
$ vfcmmap -vadapter vfchost1 -fcp fcs1
```

For non-critical systems (e.g. NIM1):

1. Map NIM1 vfchost3 adapter to physical FC adapter fcs2 on **vio1**.

```
$ vfcmmap -vadapter vfchost3 -fcp fcs2
```

2. Map NIM1 vfchost4 adapter to physical FC adapter fcs3 on **vio1**.

```
$ vfcmap -vadapter vfchost4 - fcp fcs3
```

3. Map NIM1 vfchost3 adapter to physical FC adapter fcs2 on **vio2**.

```
$ vfcmap -vadapter vfchost3 - fcp fcs2
```

4. Map NIM1 vfchost4 adapter to physical FC adapter fcs3 on **vio2**.

```
$ vfcmap -vadapter vfchost4 -fcp fcs3
```

I used the `lsmmap -all -npiv` command on each of the VIO servers to confirm that the mapping of the vfchost adapters, to the physical FC ports, was correct (as shown below).

vio1 (production LPAR):

```

Name                Physloc                ClntID ClntName            ClntOS
-----
vfchost0          U8233.E8B.XXXXXXX-V1-C66          4 LPAR4              AIX

Status:LOGGED_IN
FC name:fcs0                FC loc code:U78A0.001.XXXXXXX-P1-C3-T1
Ports logged in:5
Flags:a<LOGGED_IN,STRIP_MERGE>
VFC client name:fcs0        VFC client DRC:U8233.E8B.XXXXXXX-V6-C30-T1

Name                Physloc                ClntID ClntName            ClntOS
-----
vfchost1          U8233.E8B.XXXXXXX-V1-C67          4 LPAR4              AIX

Status:LOGGED_IN
FC name:fcs1                FC loc code:U78A0.001.XXXXXXX-P1-C3-T2
Ports logged in:5
Flags:a<LOGGED_IN,STRIP_MERGE>
VFC client name:fcs1        VFC client DRC:U8233.E8B.XXXXXXX-V6-C31-T1

```

vio1 (non-production LPAR):

```

Name                Physloc                ClntID ClntName            ClntOS
-----
vfchost3          U8233.E8B.XXXXXXX-V1-C30          3 nim1                AIX

Status:LOGGED_IN
FC name:fcs2                FC loc code:U5877.001.XXXXXXX-P1-C1-T1
Ports logged in:5
Flags:a<LOGGED_IN,STRIP_MERGE>
VFC client name:fcs0        VFC client DRC:U8233.E8B.XXXXXXX-V3-C30-T1

```

Name	Physloc	ClntID	ClntName	ClntOS
vfchost4	U8233.E8B.XXXXXXX-V1-C31	3	nim1	AIX
Status:LOGGED_IN				
FC name:fcs3		FC loc code:U5877.001.XXXXXXX-P1-C1-T2		
Ports logged in:5				
Flags:a<LOGGED_IN,STRIP_MERGE>				
VFC client name:fcs1		VFC client DRC:U8233.E8B.XXXXXXX-V3-C31-T1		

vio2 (production LPAR):

Name	Physloc	ClntID	ClntName	ClntOS
vfchost0	U8233.E8B.XXXXXXX-V2-C66	4	LPAR4	AIX
Status:LOGGED_IN				
FC name:fcs0		FC loc code:U5877.001.XXXXXXX-P1-C3-T1		
Ports logged in:5				
Flags:a<LOGGED_IN,STRIP_MERGE>				
VFC client name:fcs2		VFC client DRC:U8233.E8B.XXXXXXX-V6-C32-T1		
Name	Physloc	ClntID	ClntName	ClntOS
vfchost1	U8233.E8B.XXXXXXX-V2-C67	4	LPAR4	AIX
Status:LOGGED_IN				
FC name:fcs1		FC loc code:U5877.001.XXXXXXX-P1-C3-T2		
Ports logged in:5				
Flags:a<LOGGED_IN,STRIP_MERGE>				
VFC client name:fcs3		VFC client DRC:U8233.E8B.XXXXXXX-V6-C33-T1		

vio2 (non-production LPAR):

Name	Physloc	ClntID	ClntName	ClntOS
vfchost3	U8233.E8B.XXXXXXX-V2-C30	3	nim1	AIX
Status:LOGGED_IN				
FC name:fcs2		FC loc code:U5877.001.XXXXXXX-P1-C4-T1		
Ports logged in:5				
Flags:a<LOGGED_IN,STRIP_MERGE>				
VFC client name:fcs2		VFC client DRC:U8233.E8B.XXXXXXX-V3-C32-T1		
Name	Physloc	ClntID	ClntName	ClntOS
vfchost4	U8233.E8B.XXXXXXX-V2-C31	3	nim1	AIX
Status:LOGGED_IN				
FC name:fcs3		FC loc code:U5877.001.XXXXXXX-P1-C4-T2		
Ports logged in:5				
Flags:a<LOGGED_IN,STRIP_MERGE>				
VFC client name:fcs3		VFC client DRC:U8233.E8B.XXXXXXX-V3-C33-T1		

Fortunately, as we were using IBM XIV storage, we did not need to install additional MPIO devices drivers to support the disk. AIX supports XIV storage natively. We did, however, install some additional management utilities from the XIV host attachment package. This gave us handy tools such as **xiv_devlist** (output shown below).

```
# lsdev -Cc disk
hdisk0          Available 30-T1-01      MPIO 2810 XIV Disk
hdisk1          Available 30-T1-01      MPIO 2810 XIV Disk
hdisk2          Available 30-T1-01      MPIO 2810 XIV Disk
hdisk3          Available 30-T1-01      MPIO 2810 XIV Disk

# lslpp -l | grep xiv
xiv.hostattachment.tools  1.5.2.0  COMMITTED  Support tools for XIV
                        connectivity

# xiv_devlist

Loading disk info...

XIV Devices
-----
Device          Size      Paths  Vol Name      Vol Id  XIV Id  XIV Host
-----
/dev/hdisk1    51.5GB   16/16  nim2_ rootvg   7       7803242  nim2
-----
/dev/hdisk2    51.5GB   16/16  nim2_ nimvg    8       7803242  nim2
-----
/dev/hdisk3   103.1GB  16/16  nim2_ imgvg    9       7803242  nim2
-----

Non-XIV Devices
-----
Device  Size  Paths
-----
```

If you are planning on implementing XIV storage with AIX, I highly recommend that you take a close look at [Anthony Vandewert's blog](#) on this topic.

You may have noticed in the diagram that the VIO servers themselves boot from internal SAS drives in the 750. Each VIO server was configured with two SAS drives and a mirrored rootvg. They did not boot from SAN.

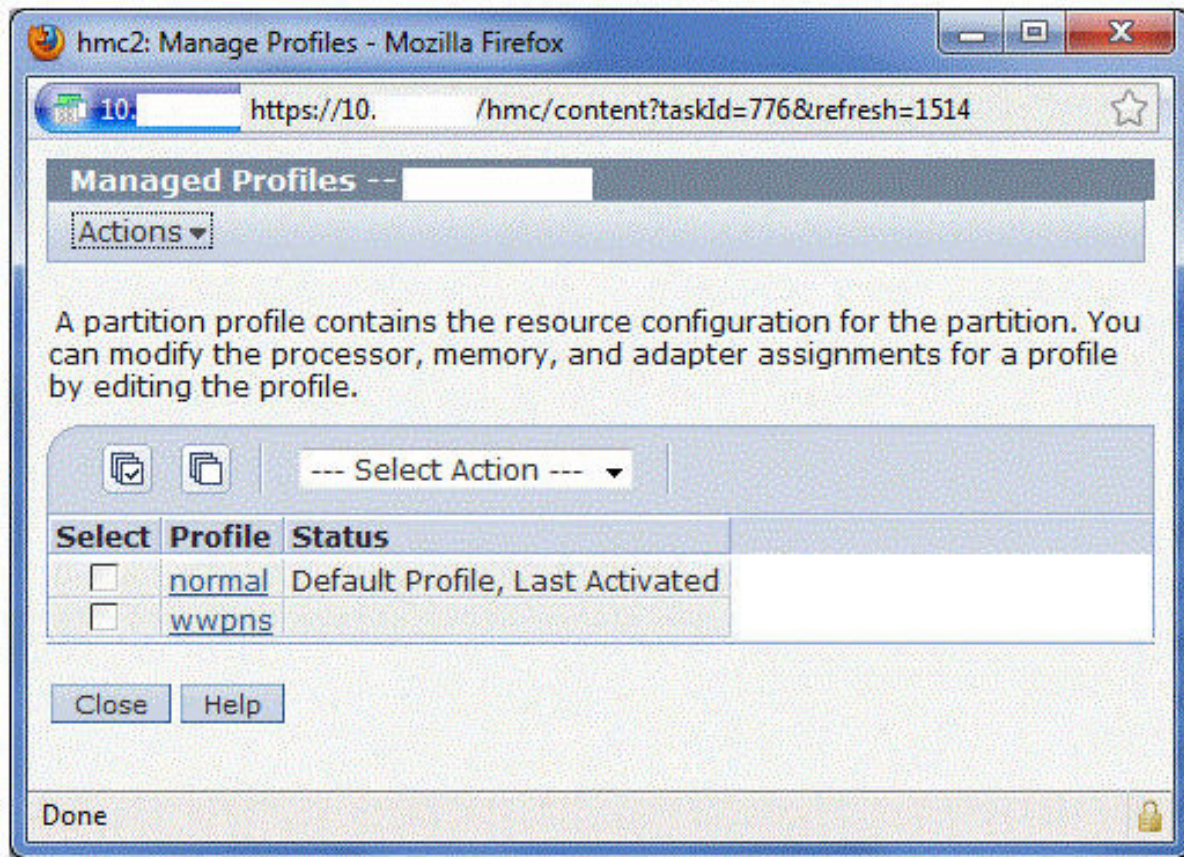
LPAR profiles

During the build of the LPARs we noticed that if we booted a new LPAR with all four of its virtual FC adapters in place, the fcsX adapter name and slot id were not in order (fcs0=slot32, fcs1=slot33, fcs3=slot30, fcs4=slot31). To prevent this from happening, we created two profiles for each LPAR.

The first profile (known as *normal*) contained the information for all four of the virtual FC adapters. The second profile (known as *wwpns*) contained only the first two virtual FC adapters that mapped to the first two physical FC ports on vio1. Using this profile to perform the LPARs first boot and to install AIX allowed the adapters to be discovered in the correct order (fcs0=slot30, fcs1=slot31). After AIX was installed and the LPAR booted, we would then re-activate the LPAR using the *normal* profile and all four virtual FC adapters.

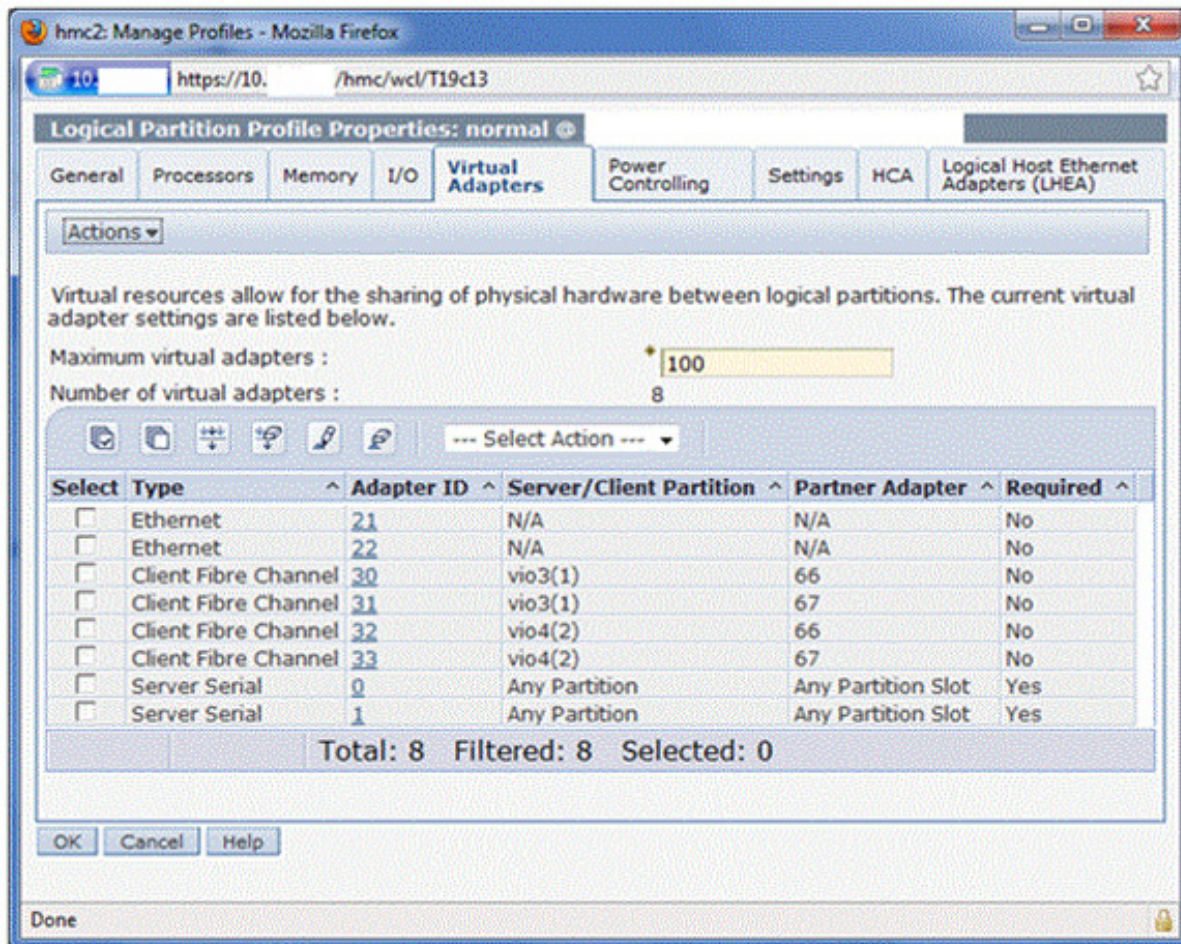
Two LPAR profiles exist for each AIX LPAR. An example is shown below.

Figure 2: LPAR profiles for virtual FC



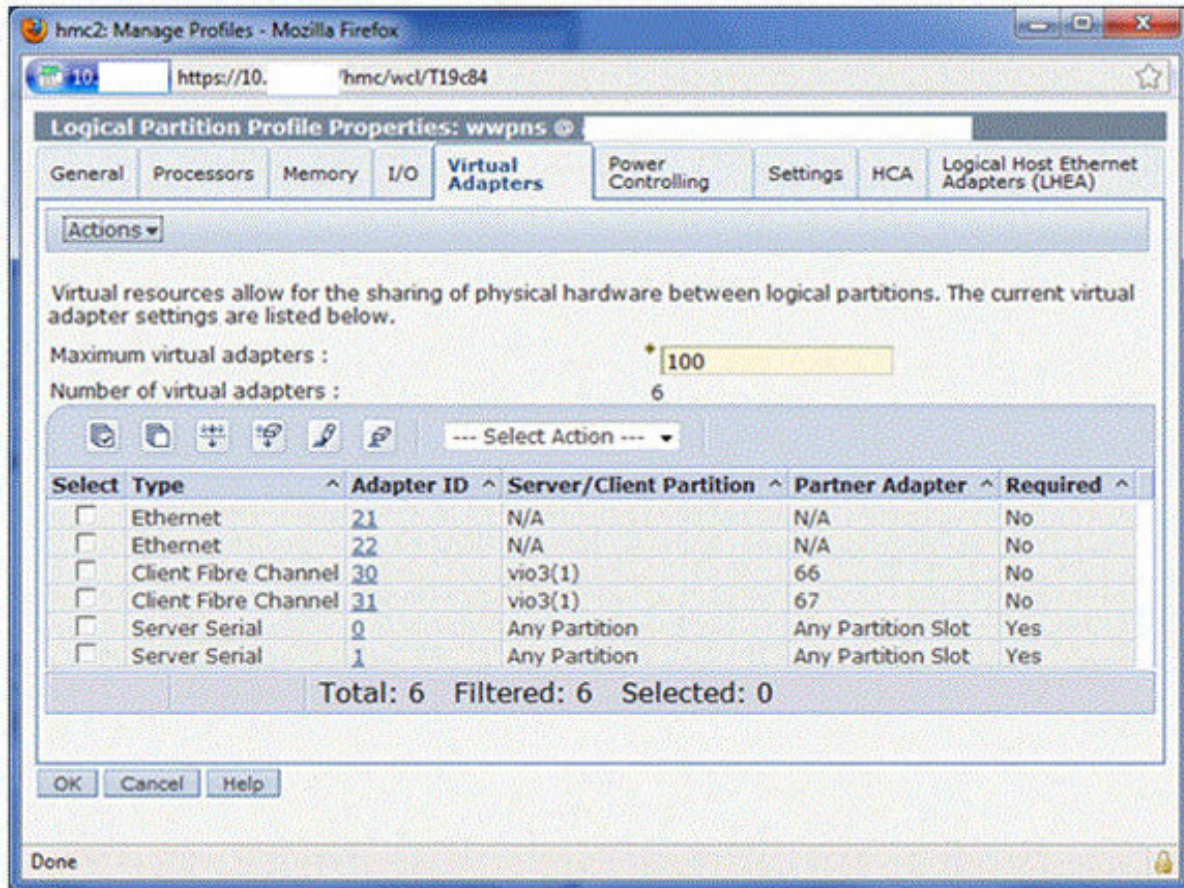
The profile named *normal* contained all of the necessary Virtual I/O devices for an LPAR (shown below). This profile was used to activate an LPAR during standard operation.

Figure 3: Profile with all virtual FC adapters used after install



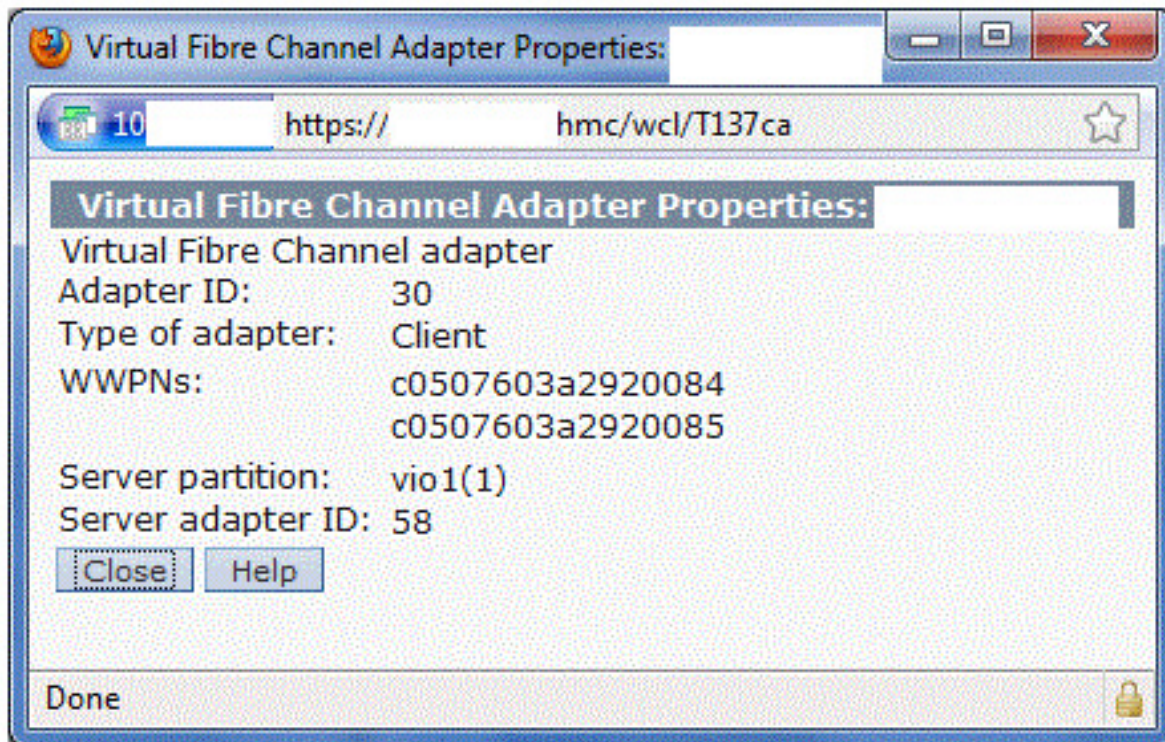
The profile named *wwpns* contained only the first two virtual FC devices for an LPAR (shown below). This profile was only used to activate an LPAR in the event that the AIX operating system needed to be reinstalled. Once the AIX installation completed successfully, the LPAR was activated again using the *normal* profile. This configured the remaining virtual FC adapters.

Figure 4: An LPAR with first two virtual FC adapters only



Also during the build process, we needed to collect a list of WWPNS for the new AIX LPARs we were installing from scratch. There were two ways we could find the WWPNS for a virtual Fibre Channel adapter on a new LPAR (for example, one that did not yet have an operating system installed). First, we started by checking the LPAR properties from the HMC (as shown below).

Figure 5: Virtual FC adapter WWPNS



To speed things up we moved to the HMC command line tool, **lssyscfg**, to display the WWPNs (as shown below).

```
hscroot@hmc1:~> lssyscfg -r prof -m 750-1 -F virtual_fc_adapters --filter lpar_names=LPAR4
""4/client/2/vio1/32/c0507603a2920084,c0507603a2920084/0"" ,
""5/client/3/vio2/32/c050760160ca0008,c050760160ca0009/0""
```

We now had a list of WWPNs for each LPAR.

```
# cat LPAR4_wwpns.txt
c0507603a292007c
c0507603a292007e
c0507603a2920078
c0507603a292007a
```

We gave these WWPNs to the SAN administrator so that he could manually "zone in" the LPARs on the SAN switches and allocate storage to each. To speed things up even more, we used **sed** to insert colons into the WWPNs. This allowed the SAN administrator to simply cut and paste the WWPNs without needing to insert colons manually.

```
# cat LPAR4_wwpns | sed 's/./&:/g;s/:$//'
```

```
c0:50:76:03:a2:92:00:7c
c0:50:76:03:a2:92:00:7e
c0:50:76:03:a2:92:00:78
```

c0:50:76:03:a2:92:00:7a

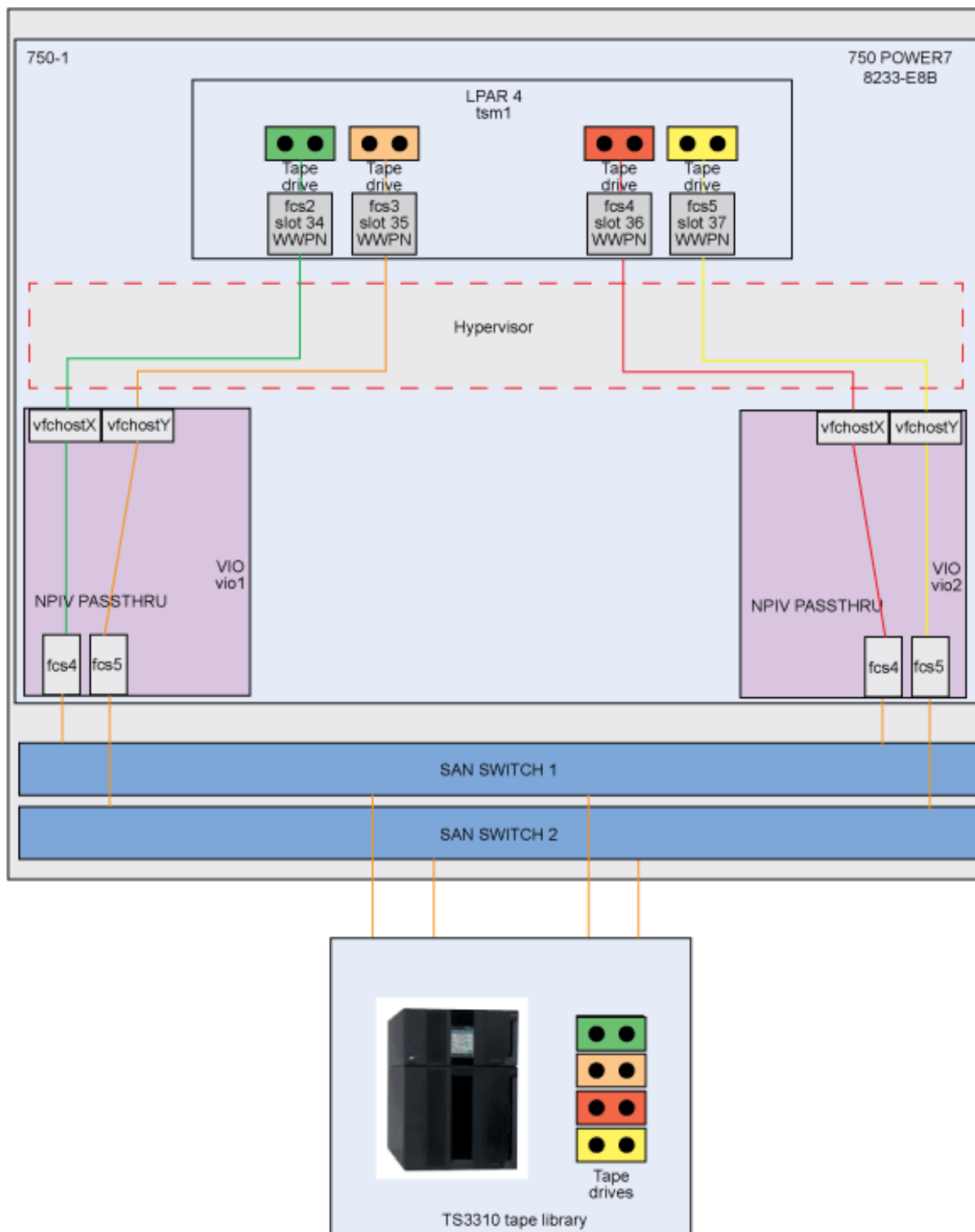
An important note here, if you plan on implementing Live Partition Mobility (LPM) with NPIV enabled systems, make sure you zone both of the WWPNs for each virtual FC adapter on the client LPAR. Remember that for each client virtual FC adapter that is created, a pair of WWPNs is generated (a primary and a secondary). Please refer to *Live Partition Mobility with Virtual Fibre Channel* in the [Resources](#) section for more information.

Virtual FC adapters for tape

Tivoli Storage Manager was the backup software used to backup and recover the systems in this new environment. Tivoli Storage Manager would use a TS3310 tape library, as well as disk storage pools to backup client data. In this environment, we chose to use virtual FC adapters to connect the tape library to Tivoli Storage Manager. This also gave us the capability to assign the tape devices to any LPAR, without moving the physical adapters from one LPAR to another, should the need arise in the future. As I mentioned earlier, there were three 2-port 8GB FC adapters assigned to each VIOS. Two adapters were used for disk and the third would be used exclusively for tape.

The following diagram shows that physical FC ports, fcs4 and fcs5, in each VIOS would be used for tape connectivity. It also shows that each of the 4 tape drives would be zoned to a specific virtual FC adapter in the Tivoli Storage Manager LPAR.

Figure 6. Tape drive connectivity via virtual FC



(View a [larger version](#) of Figure 6.)

The Tivoli Storage Manager LPAR was initially configured with virtual FC adapters for connectivity to XIV disk only. As shown in the **lspath** output below, fcs0 through

fcs3 are used exclusively for access to disk only.

```
# lsdev -Cc adapter | grep fcs
fcs0 Available 30-T1 Virtual Fibre Channel Client Adapter
fcs1 Available 31-T1 Virtual Fibre Channel Client Adapter
fcs2 Available 32-T1 Virtual Fibre Channel Client Adapter
fcs3 Available 33-T1 Virtual Fibre Channel Client Adapter

# lspath
Enabled hdisk0 fscsi0
Enabled hdisk0 fscsi0
Enabled hdisk0 fscsi0
Enabled hdisk0 fscsi0
Enabled hdisk0 fscsi1
Enabled hdisk0 fscsi1
Enabled hdisk0 fscsi1
Enabled hdisk0 fscsi1
Enabled hdisk0 fscsi2
Enabled hdisk0 fscsi2
Enabled hdisk0 fscsi2
Enabled hdisk0 fscsi2
Enabled hdisk0 fscsi3
Enabled hdisk0 fscsi3
Enabled hdisk0 fscsi3
Enabled hdisk0 fscsi3
..etc.. for the other disks on the system
```

To connect to the tape drives, we configured four additional virtual FC adapters for the LPAR. First, we ensured that the physical adapters were available and had fabric connectivity. On both VIOS, we used the `lsnports` command to determine the state of the adapters and their NPIV capability. As shown in the following output, the physical adapter's `fcs4` and `fcs5` were both available and NPIV ready. There was a 1 in the *fabric* column. If it was zero then the adapter may not be connected to an NPIV capable SAN.

```
$ lsnports
name physloc fabric tports aports swwpns awwpns
fcs0 U78A0.001.DNWK4W9-P1-C3-T1 1 64 52 2048 1988
fcs1 U78A0.001.DNWK4W9-P1-C3-T2 1 64 52 2048 1988
fcs2 U5877.001.0084548-P1-C1-T1 1 64 61 2048 2033
fcs3 U5877.001.0084548-P1-C1-T2 1 64 61 2048 2033
fcs4 U5877.001.0084548-P1-C2-T1 1 64 64 2048 2048
fcs5 U5877.001.0084548-P1-C2-T2 1 64 64 2048 2048
```

When I initially checked the state of the adapters on both VIOS, I encountered the following output from `lsnports`:

```
$ lsnports
name physloc fabric tports aports swwpns awwpns
fcs0 U78A0.001.DNWK4W9-P1-C3-T1 1 64 52 2048 1988
fcs1 U78A0.001.DNWK4W9-P1-C3-T2 1 64 52 2048 1988
fcs2 U5877.001.0084548-P1-C1-T1 1 64 61 2048 2033
fcs3 U5877.001.0084548-P1-C1-T2 1 64 61 2048 2033
fcs4 U5877.001.0084548-P1-C2-T1 0 64 64 2048 2048
```

As you can see, only the fcs4 adapter was discovered; the fabric value for fcs4 was 0 and fcs5 was missing. Both of these issues were the result of physical connectivity issues to the SAN. The cables were unplugged and/or they had a loopback adapter plugged into the interface. The error report indicated link errors on fcs4 but not for fcs5.

```
$ errlog
IDENTIFIER  TIMESTAMP  T C RESOURCE_NAME  DESCRIPTION
7BFEEA1F  0502104011  T H fcs4  LINK ERROR
```

Once the ports were physically connected to the SAN switches, I removed the entry for fcs4 from the ODM (as shown below) and then ran cfgmgr on the VIOS.

```
$ r oem
oem_setup_env
# rmdev -dRl fcs4
fcnet4 deleted
sfwcomm4 deleted
fscsi4 deleted
fcs4 deleted
# cfgmgr
# exit
$
```

Then both fcs4 and fcs5 were discovered and configured correctly.

```
$ lsnports
name  physloc          fabric      tports  aports  swwpns  awwpns
fcs0  U78A0.001.DNWK4W9-P1-C3-T1  1        64      52      2048   1988
fcs1  U78A0.001.DNWK4W9-P1-C3-T2  1        64      52      2048   1988
fcs2  U5877.001.0084548-P1-C1-T1  1        64      61      2048   2033
fcs3  U5877.001.0084548-P1-C1-T2  1        64      61      2048   2033
fcs4  U5877.001.0084548-P1-C2-T1  1        64      64      2048   2048
fcs5  U5877.001.0084548-P1-C2-T2  1        64      64      2048   2048
```

The Tivoli Storage Manager LPARs dedicated virtual FC adapters, for tape, appeared as fcs4, fcs5, fcs6 and fcs7. The plan was for fcs4 on tsm1 to map to fcs4 on vio1, fcs5 to map to fcs5 on vio1, fcs6 to map to fcs4 on vio2, and fcs7 to map to fcs5 on vio2.

The virtual adapter slot configuration was as follows:

```
LPAR: tsm1  VIOS: vio1
U8233.E8B.06XXXXXX-V4-C34-T1 > U8233.E8B.06XXXXXX-V1-C60
U8233.E8B.06XXXXXX-V4-C35-T1 > U8233.E8B.06XXXXXX-V1-C61

LPAR: tsm1  VIOS: vio2
U8233.E8B.06XXXXXX-V4-C36-T1 > U8233.E8B.06XXXXXX-V2-C60
U8233.E8B.06XXXXXX-V4-C37-T1 > U8233.E8B.06XXXXXX-V2-C61
```


We created two new virtual FC host (vfchost) adapters on vio1 and two new vfchost adapters on vio2. This was done by updating the profile for both VIOS (on the HMC) with the new adapters and then adding them with a DLPAR operation on each VIOS. Once we had run the `cfgdev` command on each VIOS to bring in the new vfchost adapters, we needed to map them to the physical FC ports.

Using the `vfcmmap` command on each of the VIOS, we mapped the physical ports to the virtual host adapters as follows:

1. Map tsm1 vfchost60 adapter to physical FC adapter fcs4 on **vio1**.

```
$ vfcmmap -vadapter vfchost60 -fcp fcs4
```

2. Map tsm1 vfchost61 adapter to physical FC adapter fcs5 on **vio1**.

```
$ vfcmmap -vadapter vfchost61 - fcp fcs5
```

3. Map tsm1 vfchost60 adapter to physical FC adapter fcs4 on **vio2**.

```
$ vfcmmap -vadapter vfchost60 - fcp fcs4
```

4. Map tsm1 vfchost61 adapter to physical FC adapter fcs5 on **vio2**.

```
$ vfcmmap -vadapter vfchost61 -fcp fcs5
```

Next we used DLPAR (using the following [procedure](#)) to update the client LPAR with four new virtual FC adapters. **Please make sure you read the procedure on adding a virtual FC adapter to client LPAR. If care is not taken, the WWPNs for a client LPAR can be lost, which can result in loss of connectivity to your SAN storage.** You may also want to review the HMC's `chsyscfg` command, as it is possible to use this command to modify WWPNs for an LPAR.

After running the `cfgmgr` command on the LPAR, we confirmed we had four new virtual FC adapters. We ensured that we saved the LPARs *current* configuration, as outlined in the procedure.

```
# lsdev -Cc adapter | grep fcs
fcs0 Available 30-T1 Virtual Fibre Channel Client Adapter
fcs1 Available 31-T1 Virtual Fibre Channel Client Adapter
fcs2 Available 32-T1 Virtual Fibre Channel Client Adapter
fcs3 Available 33-T1 Virtual Fibre Channel Client Adapter
fcs4 Available 34-T1 Virtual Fibre Channel Client Adapter
fcs5 Available 35-T1 Virtual Fibre Channel Client Adapter
```

```
fcs6 Available 36-T1 Virtual Fibre Channel Client Adapter
fcs7 Available 37-T1 Virtual Fibre Channel Client Adapter
```

On both VIOS, we confirmed that the physical to virtual mapping on the FC adapters was correct using the `lsmmap -all -npiv` command. Also checking that client LPAR had successfully logged into the SAN by noting the `Status: LOGGED_IN` entry in the `lsmmap` output for each adapter.

```
vio1:
Name Physloc ClntID ClntName ClntOS
-----
vfchost60 U8233.E8B.06XXXXX-V1-C60 6 tsm1 AIX

Status:LOGGED_IN
FC name:fcs4 FC loc code:U5877.001.0084548-P1-C2-T1
Ports logged in:1
Flags:a<LOGGED_IN,STRIP_MERGE>
VFC client name:fcs4 VFC client DRC:U8233.E8B.06XXXXX-V4-C34-T1

Name Physloc ClntID ClntName ClntOS
-----
vfchost61 U8233.E8B.06XXXXX-V1-C61 6 tsm1 AIX

Status:LOGGED_IN
FC name:fcs5 FC loc code:U5877.001.0084548-P1-C2-T2
Ports logged in:1
Flags:a<LOGGED_IN,STRIP_MERGE>
VFC client name:fcs5 VFC client DRC:U8233.E8B.06XXXXX-V4-C35-T1

vio2:
Name Physloc ClntID ClntName ClntOS
-----
vfchost60 U8233.E8B.06XXXXX-V2-C60 6 tsm1 AIX

Status:LOGGED_IN
FC name:fcs4 FC loc code:U5877.001.0084548-P1-C5-T1
Ports logged in:1
Flags:a<LOGGED_IN,STRIP_MERGE>
VFC client name:fcs6 VFC client DRC:U8233.E8B.06XXXXX-V4-C36-T1

Name Physloc ClntID ClntName ClntOS
-----
vfchost61 U8233.E8B.06XXXXX-V2-C61 6 tsm1 AIX

Status:LOGGED_IN
FC name:fcs5 FC loc code:U5877.001.0084548-P1-C5-T2
Ports logged in:1
Flags:a<LOGGED_IN,STRIP_MERGE>
VFC client name:fcs7 VFC client DRC:U8233.E8B.06XXXXX-V4-C37-T1
```

We were able to capture the WWPNs for the new adapters at this point. This information was required to zone the tape drives to the system.

```
# for i in 4 5 6 7
> do
> echo fcs$i
> lscfg -vpl fcs$i | grep Net
> echo
> done
```

```
fcs4
  Network Address.....C0507603A2720087

fcs5
  Network Address.....C0507603A272008B

fcs6
  Network Address.....C0507603A272008C

fcs7
  Network Address.....C0507603A272008D
```

The IBM Atape device drivers were installed prior to zoning in the TS3310 tape drives.

```
# lslpp -l | grep -i atape
Atape.driver 12.2.4.0 COMMITTED IBM AIX Enhanced Tape and
```

Then, once the drives had been zoned to the new WWPNS, we ran **cfgmgr** on the Tivoli Storage Manager LPAR to configure the tape drives.

```
# lsdev -Cc tape

# cfgmgr
# lsdev -Cc tape
rmt0 Available 34-T1-01-PRI IBM 3580 Ultrium Tape Drive (FCP)
rmt1 Available 34-T1-01-PRI IBM 3580 Ultrium Tape Drive (FCP)
rmt2 Available 35-T1-01-ALT IBM 3580 Ultrium Tape Drive (FCP)
rmt3 Available 35-T1-01-ALT IBM 3580 Ultrium Tape Drive (FCP)
rmt4 Available 36-T1-01-PRI IBM 3580 Ultrium Tape Drive (FCP)
rmt5 Available 36-T1-01-PRI IBM 3580 Ultrium Tape Drive (FCP)
rmt6 Available 37-T1-01-ALT IBM 3580 Ultrium Tape Drive (FCP)
rmt7 Available 37-T1-01-ALT IBM 3580 Ultrium Tape Drive (FCP)
smc0 Available 34-T1-01-PRI IBM 3576 Library Medium Changer (FCP)
smc1 Available 35-T1-01-ALT IBM 3576 Library Medium Changer (FCP)
smc2 Available 37-T1-01-ALT IBM 3576 Library Medium Changer (FCP)
```

Our new tape drives were now available to Tivoli Storage Manager.

Monitoring virtual FC adapters

Apparently the **viostat** command on the VIO server allows you to monitor I/O traffic on the vfchost adapters (as shown in the following example).

```
$ viostat -adapter vfchost3
System configuration: lcpu=8 drives=1 ent=0.50 paths=4 vdisks=20 tapes=0

tty: tin tout  avg-cpu: % user % sys % idle % iowait physc %
entc
0.0 0.2 0.0 0.2 99.8 0.0 0.0
0.4

Adapter: Kbps tps Kb_read Kb_wrtn
```

```

fcs1 2.5 0.4 199214 249268

Adapter: Kbps tps Kb_read Kb_wrtn
fcs2 0.0 0.0 0 0

Vadapter: Kbps tps bkread bkwrtn
vfchost4 0.0 0.0 0.0 0.0

Vadapter: Kbps tps bkread bkwrtn
vfchost6 0.0 0.0 0.0 0.0

Vadapter: Kbps tps bkread bkwrtn
vfchost5 0.0 0.0 0.0 0.0

Vadapter: Kbps tps bkread bkwrtn
vfchost0 0.0 0.0 0.0 0.0

Vadapter: Kbps tps bkread bkwrtn
vfchost3 0.0 0.0 0.0 0.0

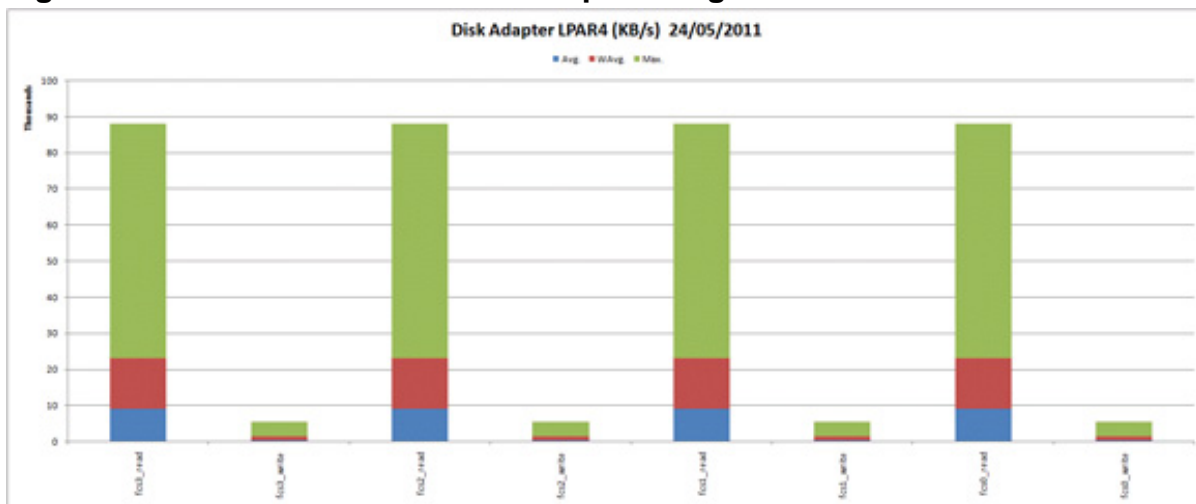
Vadapter: Kbps tps bkread bkwrtn
vfchost2 0.0 0.0 0.0 0.0

Vadapter: Kbps tps bkread bkwrtn
vfchost1 0.0 0.0 0.0 0.0
    
```

I must admit I had limited success using this tool to monitor I/O on these devices. I am yet to discover why this tool did not report any statistics for any of my vfchost adapters. Perhaps it was an issue with the level of VIOS code we were running?

Fortunately, nmon captures and reports on virtual FC adapter performance statistics on the client LPAR. This is nothing new, as nmon has always captured FC adapter information, but it is good to know that nmon can record the data for both virtual and physical FC adapters.

Figure 7. nmon data for virtual FC adapter usage



(View a [larger version](#) of Figure 7.)

The **fcstat** command can be used on the client LPARs to monitor performance statistics relating to buffer usage and overflows on the adapters. For example, the

following output indicated that we needed to tune some of the settings on our virtual FC adapters. In particular the following attributes were modified, *num_cmd_elems* and *max_xfer_size*.

```
# fcstat fcs0 | grep -p DMA | grep -p 'FC SCSI'
FC SCSI Adapter Driver Information
No DMA Resource Count: 580
No Adapter Elements Count: 0
No Command Resource Count: 6093967

# fcstat fcs1 | grep -p DMA | grep -p 'FC SCSI'
FC SCSI Adapter Driver Information
No DMA Resource Count: 386
No Adapter Elements Count: 0
No Command Resource Count: 6132098

# fcstat fcs2 | grep -p DMA | grep -p 'FC SCSI'
FC SCSI Adapter Driver Information
No DMA Resource Count: 222
No Adapter Elements Count: 0
No Command Resource Count: 6336080

# fcstat fcs3 | grep -p DMA | grep -p 'FC SCSI'
FC SCSI Adapter Driver Information
No DMA Resource Count: 875
No Adapter Elements Count: 0
No Command Resource Count: 6425427
```

We also found buffer issues (via the **fcstat** command) on the physical adapters on the VIO servers. We tuned the FC adapters on the VIO servers to match the settings on the client LPARs, such as *max_xfer_size=0x200000* and *num_cmd_elems=2048*.

The **fcstat** command will report a value of *UNKNOWN* for some attributes of a virtual FC adapter. Because it is a virtual adapter, it does not contain any information relating to the physical adapter attributes, such as firmware level information or supported port speeds.

```
# fcstat fcs0
FIBRE CHANNEL STATISTICS REPORT: fcs0
Device Type: FC Adapter (adapter/vdevice/IBM,vfc-client)
Serial Number: UNKNOWN
Option ROM Version: UNKNOWN
Firmware Version: UNKNOWN
World Wide Node Name: 0xC0507603A202007c
World Wide Port Name: 0xC0507603A202007e
FC-4 TYPES:
Supported: 0x000001000000000000000000000000000000000000000000000000000000000000
Active: 0x000001000000000000000000000000000000000000000000000000000000000000
Class of Service: 3
Port Speed (supported): UNKNOWN
Port Speed (running): 8 GBIT
Port FC ID: 0x5D061D
```

Conclusion

In all that describes my experience with NPIV, Power Systems, Virtual I/O and AIX. I hope you have enjoyed reading this article. Of course, as they say, "*there's always more than one way to skin a cat*"! So please feel free to contact me and share your experiences with this technology, I'd like to hear your thoughts and experiences.

Resources

Learn

- Please refer to the following links for more information relating to virtual FC configuration in a VIOS environment.
 - [VIOS for AIX Administrators](#)
 - [IBM PowerVM Virtualization Managing and Monitoring](#)
 - [PowerVM – Dynamically adding a Virtual Fibre Channel adapter to a client partition](#)
 - [Implementing Live Mobility with Virtual Fibre Channel](#)

Get products and technologies

- [Try out IBM software](#) for free. Download a trial version, log into an online trial, work with a product in a sandbox environment, or access it through the cloud. Choose from over 100 IBM product trials.

Discuss

- Follow [developerWorks on Twitter](#).
- Participate in [developerWorks blogs](#) and get involved in the developerWorks community.
- Get involved in the [My developerWorks community](#).
- Participate in the AIX and UNIX® forums:
 - [AIX Forum](#)
 - [AIX Forum for developers](#)
 - [Cluster Systems Management](#)
 - [Performance Tools Forum](#)
 - [Virtualization Forum](#)
 - [More AIX and UNIX Forums](#)

About the author

Chris Gibson



Chris Gibson is an AIX specialist located in Melbourne, Australia. He is an IBM CATE, System p platform and AIX 5L, and a co-author of several IBM Redbooks on AIX. You contact Chris at cg@gibsonnet.net, on Twitter [@cgibbo](https://twitter.com/cgibbo) or his [AIX blog](#).