Using Virtual Switches in PowerVM to Drive Maximum Value of 10 Gb Ethernet

by
Glenn E. Miller
Certified IT Specialist
Power Systems, AIX and PowerHA
IBM Corporation

and
Kris Speetjens
IT Architect
Nobius Solutions, Incorporated

September 10, 2010

Table of Contents

Table of Contents....2
Trademarks....3
List of Tables and Figures....4
Background....5
Working Toward a Bullet Proof System....8
1. Protecting Against Failure of Virtual I/O Servers....8
2. Protecting Against Failure of Ethernet Switches, Cables, or Adapters....8
3. Protecting Against Misconfiguration of a Virtual I/O Server....9
4. Protecting Against Misconfiguration of Ethernet switches....9
Implementation....10
Part 1: Virtual Switch Creation....10
Part 2: Virtual Ethernet Server Adapter....13
Part 3: Shared Ethernet Adapter....15
Part 4: Virtual Client Network Interface....17
Part 5: Cisco Switch Configuration....22
Conclusion....23
References....24

Trademarks

List of Tables and Figures

Figure 1: Classic NIB Implementation....5
Figure 2: Classic Shared Ethernet Adapter Failover Implementation....6
Figure 3: Multiple Virtual Switch Example....7
Table 1: SEA Failover vs. vSwitch Comparison....9
Figure 4: Creating the virtual switch (1)....10
Figure 5: Creating the virtual switch (2)....11
Figure 6: Creating the virtual switch (3)....11
Figure 7: Creating the virtual switch (4)....12
Figure 8: Creating the virtual switch (5)....12
Figure 9: Creating the virtual Ethernet server adapter (1)....13
Figure 10: Creating the virtual Ethernet server adapter (2)....13
Figure 11: Creating the virtual Ethernet server adapter (3)....14
Figure 12: Creating the virtual Ethernet server adapter (4)....15
Figure 13: Creating the virtual Ethernet server adapter (5)....15
Figure 14: Creating the VLAN Tag Devices....16
Figure 15: Viewing the VLAN Tag Devices....17
Figure 16: Viewing the VLAN Tag Attributes....17
Figure 17: Viewing the Client's Virtual Adapters....18
Figure 18: Configuring the Client's First Virtual Ethernet Adapter Properties....18
Figure 19: Configuring the Client's Second Virtual Ethernet Adapter Properties....19
Figure 20: Checking configuration of Virtual Ethernet Adapter....19
Figure 21: Checking configuration of Virtual Ethernet Adapter....20
Figure 22: Configuring the NIB device (1)....20
Figure 23: Configuring the NIB device (2)....21
Figure 24: Examining the Error Log....21
Figure 25: Example Cisco Port Configuration....22
Figure 26: As-Built Diagram of Configuration....23

Background

What are virtual switches? Why do we need them? Hopefully, these questions and many more will be answered by the time you finish reading this paper. To start, it would be helpful to provide a little background. We are making the assumption that the reader has a basic understanding of IBM's PowerVM product.

The goal of most System Administrators is to do everything possible in the design and implementation of their server to provide redundancy and failover protection. Thus, if a hardware component were to fail, the system would still be operational. Obviously, to account for the server itself possibly failing, you can add a redundant server into the design thereby providing High Availability, but that's a topic for another paper. In light of this, we're going to discuss some methodologies that we use to provide redundancy within a server, specifically for Ethernet. Further, we're going to discuss a new method that you need to consider, particularly if implementing 10 Gb Ethernet.

When configuring a Power System with virtualization, typically, we implement two Virtual I/O Servers. This is the first step to satisfying the requirement for redundancy. Each Virtual I/O Server is allocated at least one physical Ethernet adapter. To provide even better redundancy, we should consider adding a second Ethernet adapter to each Virtual I/O Server. Typically this would be done by creating a “Network Interface Backup” (NIB) Etherchannel device. You could substitute a Link aggregation Etherchannel for the NIB device, but to do that, both Ethernet adapters from the Virtual I/O Server have to be connected to the same physical switch, which is a potential single point of failure, so you may still wish to provide a backup adapter and NIB to protect against that.

Figure 1: Classic NIB Implementation

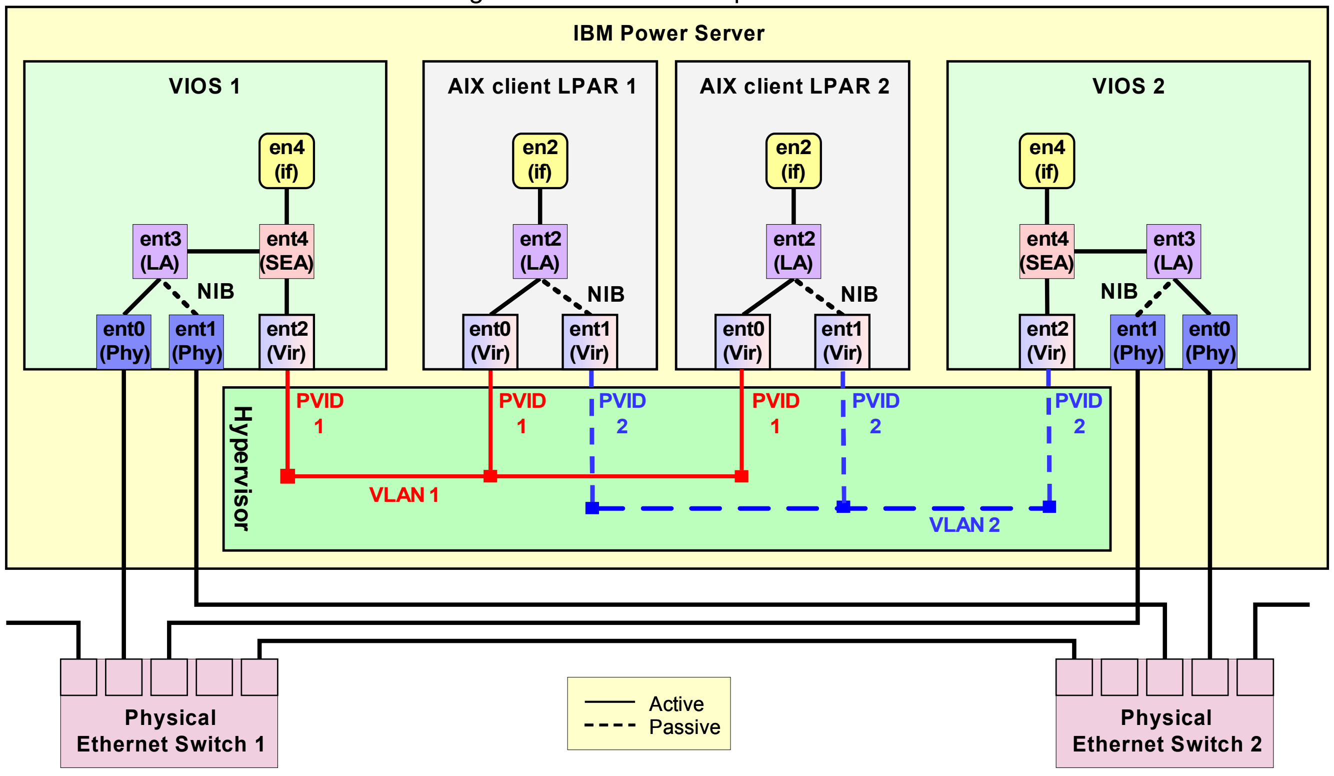


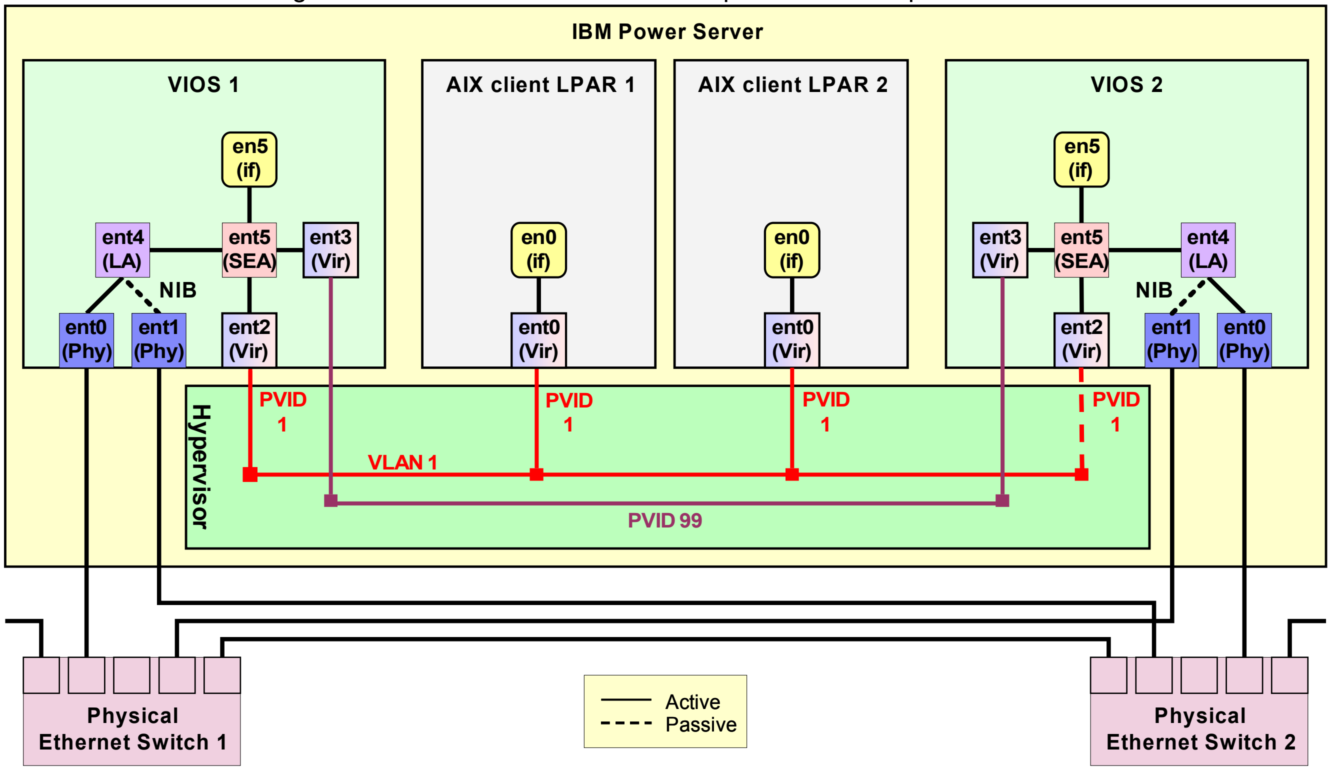
Figure 1 depicts this design. Notice that the Power Hypervisor is acting in the capacity of a single virtual Ethernet switch (ethernet0 is the default name for this switch). Additionally, that there are two virtual Ethernet adapters in each client Logical Partition (LPAR). Those adapters are connected to the virtual switch and consequently form a connection to the virtual adapters in each of the Virtual I/O Servers. However, the second virtual Ethernet adapter in each client and the virtual Ethernet adapter in the second Virtual I/O Server are configured to reside on a second virtual network, VLAN2.

If both Virtual I/O Servers were connected to the same VLAN through the same virtual switch (ethernet0), transmissions from the client could use either Virtual I/O Server to access the external network. In the best case scenario this would cause “port flapping” as spanning tree recalculates to prevent bridging loops or in a worse case scenario spanning tree would be turned off and you would have a partial network outage caused by a layer-2 broadcast loop. Therefore, the use of separate VLANs guarantees that traffic generated in the clients will only use one path at a time to the physical network.

However, the fact that the client LPAR is connected to two separate VLANs prevents the use of VLAN tagging. VLAN tagging cannot be used in this scenario because trunks are required to support VLANs and two trunks would mean that there are two links that could carry the traffic creating the same situation as described previously. This is because you would need both adapters in the same VLAN for NIB to function. Remember, the reason VLAN tagging cannot be supported in this case is because you can’t have one (virtual) switch with two connections.

To get around this problem, a different configuration needs to be used, namely, Shared Ethernet Adapter (SEA) Failover. Figure 2 shows a typically used example of this configuration. Here the clients only need a single connection to the virtual switch because both Virtual I/O Servers are connected to the same VLAN. An additional component required for this to work is the control channel that is established between the Virtual I/O Servers. The purpose of the control channel is for the Virtual I/O Servers to communicate with each other similar to heartbeats used in High Availability clusters. Essentially, one Virtual I/O Server is acting as the primary and the other as secondary route to the physical network. The Server in the secondary role prevents traffic from going out to the physical network and consequently from forming a network loop. Only in the event of a failure of the primary Server's access to the external network will the secondary allow traffic to go out.

Figure 2: Classic Shared Ethernet Adapter Failover Implementation



Since the two Virtual I/O Servers can be connected to the same VLAN, the use of VLAN tagging is permitted in this scenario.

Something that we haven't pointed out thus far in the discussion is the fact that redundancy does have it's drawbacks. The backup adapter is fundamentally unused unless a failure occurs. In the example depicted in Figure 2, there are three physical adapters and their corresponding Ethernet switch ports that are never used except when a failure condition occurs. These ports have associated costs. Within the more common 1 Gb environment, it's not too drastic. However, in the 10 Gb environment it's vastly different. One customer estimated that it cost them $16,000 for each 10 Gb/s connection provided in their data center, taking into account the cost of the Ethernet adapter, cabling and the proportionate cost of the chassis, blade and port of the Ethernet switch. Obviously, 10 Gb connectivity is going to be a necessity in the near future as customers continue to consolidate more and more workloads onto smaller, much more powerful systems. However, it it may be difficult to justify 40 Gb worth of bandwidth when only 10 Gb will be utilized.

In the NIB design shown earlier in Figure 1, the second Virtual I/O Server could be actively used by some of the client LPARs if their primary path was set to use ent1, or if they were manually failed over from ent0, however we are still left with the inability to use VLAN tagging. What was needed is the ability to use NIB in the client with a single VLAN to both Virtual I/O Servers. A feature built into the Power Hypervisor that delivers this capability is the ability to create more than one IEEE VLAN compatible virtual switch.

Figure 3: Multiple Virtual Switch Example

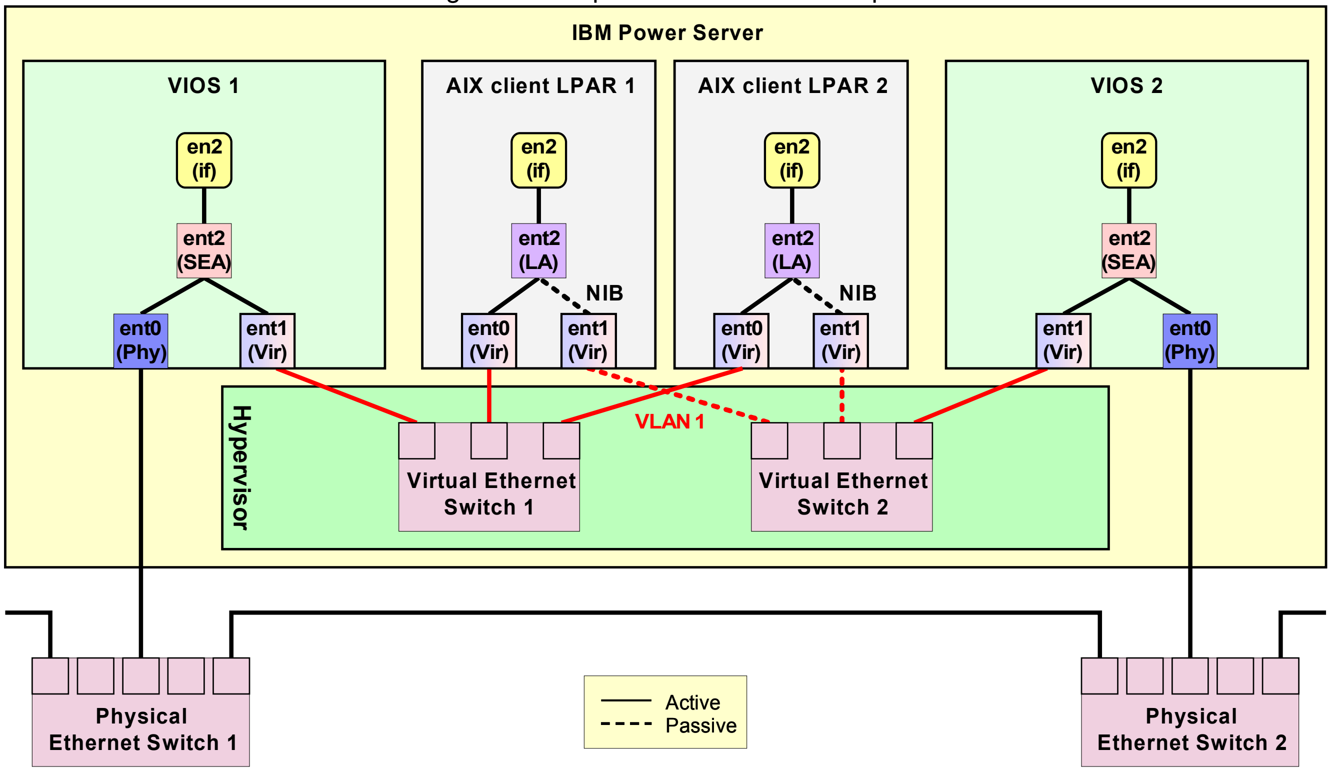


Figure 3 shows an example of how the system could be redesigned to use multiple virtual switches. Note that there is a separate virtual switch for each Virtual I/O Server. In the client configuration each adapter is "attached" to a separate virtual switch which guarantees it can only use that particular Virtual I/O Server for external access. Since NIB is used in the client there is only one adapter active in the client and consequently it only accesses the external network through one Virtual I/O Server. The Port Virtual LAN Identifier (PVID) doesn't need to be specified because SEA failover is not used here. The point of the PVID is to help determine which link spanning tree should disable to avoid loops. Since there are two virtual switches both links can remain active without the need for spanning tree to shut one down to protect the network from loops as there will never be traffic sent down two paths with the same source MAC address. Therefore there can be multiple adapters in the same VLAN and VLAN tagging can be permitted because MAC address stability is guaranteed.

A significant benefit to this design is that both Virtual I/O Servers can be active at the same time. Of course, each individual client LPAR is only using one, but half of the clients could be configured to use Virtual I/O Server 1 and the other half to use Virtual I/O Server 2 as their primary paths. Each client would failover to it's respective secondary path in the case that it's primary path was lost. So the customer's investment in hardware is more effectively utilized.

Working Toward a Bullet Proof System

As the level of workload consolidation increases so does the need for fault tolerance to protect the environment. This section attempts to explain how the Virtual Switch design creates the optimum solution in terms of fault tolerance and system flexibility. To understand the design choices made in both the “classic” Shared Ethernet Adapter Failover and new vSwitch design one has to look at the failover scenarios that are trying to be avoided. The four primary failure scenarios we are trying to protect against are (1) failure of VIO Server, (2) failure of Ethernet switches, cables, or adapters, (3) misconfiguration of VIO Servers and (4) misconfiguration of Ethernet switches.

1. Protecting Against Failure of Virtual I/O Servers

Protection against this scenario is accomplished by configuring two Virtual I/O servers on each Power System frame and assigning resources to the Virtual I/O clients from both Virtual I/O servers. The “classic” design that allows use of VLAN tagging (Figure 2) uses a control channel to allow the Virtual I/O servers to detect a failure and handle Ethernet traffic accordingly. The vSwitch design handles this at the client level by pinging external resources and failing over the Client NIB Etherchannel when a threshold of failed pings is reached.

The “classic” designs' advantages are that it requires no configuration at the Virtual I/O client level and all clients can be migrated from one Virtual I/O server to another with the execution of one command on the Virtual I/O server during system maintenance. The disadvantages of the “classic” design is that only one Virtual I/O server is carrying Ethernet traffic at any time which means a systems is only utilizing 50% of it's available bandwidth at any time. It also means that there is no way to test if the failover link is correct without failing over every Virtual I/O client on a frame.

The vSwitch designs' advantages are that it allows both Virtual I/O servers to carry Ethernet traffic at the same time. This means that administrators are given more granular control over moving Ethernet traffic from one Virtual I/O server to another as well as utilizing a higher percentage of bandwidth during normal operations. The disadvantage of the vSwitch design is that it requires every Virtual I/O client (which uses Network Interface Backup to verify path integrity) to ping an address outside of the frame to test for failures.

2. Protecting Against Failure of Ethernet Switches, Cables, or Adapters

Protection against this scenario is accomplished by the same components provided to protect against a Virtual I/O Server Failure. To protect against a physical failure both designs require at least two Ethernet adapters, one controlled by each Virtual I/O server. The major differences between the two scenarios is that the vSwitch design will already have all links carrying traffic. This means that in the event of a failure fewer systems will have to failover.

3. Protecting Against Misconfiguration of a Virtual I/O Server

Protection against this scenario is very difficult for both environments; however, the vSwitch design provides several advantages over the “classic” design. Since the vSwitch design requires each client to test network connectivity, the clients are already positioned to survive a misconfiguration on their active Virtual I/O server. Also since both Virtual I/O servers have active network connections, administrators can test network configurations on secondary Virtual I/O server through methods such as testing network connectivity of test boxes and/or assigning IP addresses to the VLAN interfaces in the Virtual I/O server and testing network connectivity from there. There is no way to test configuration changes in the “classic” example without failing all Virtual I/O clients to it since the control channel and spanning-tree ensure the backup link is not active.

4. Protecting Against Misconfiguration of Ethernet switches

It is the vSwitch designs' ability to recover from switch misconfiguration that makes the design very appealing especially when considering the higher cost of the high throughput adapters. It is important to realize that with the “classic” design networking will only failover between Virtual I/O servers if physical connectivity is lost on the primary Virtual I/O server. If the link is up, but not forwarding traffic due to either a misconfiguration or a partial failure all traffic to the Virtual I/O clients will be lost.

To overcome this administrators add multiple adapters to each Virtual I/O server and create NIB interfaces which are set to ping an external address. In the event one link does not respond to the ping tests, the NIB interfaces will failover to its backup adapter. This now means that each Virtual I/O server pair needs four adapters to fully protect the network from failure. However; there are still problems with this design that make it less than optimal. If the administrator decides to not use VLAN trunks, then they will need four adapters per VLAN to provide full redundancy. This is obviously cost prohibitive at the 10 Gb/s, but it also presents design challenges at the 1 Gb/s design as well. If the administrator decides to use VLAN trunks there is another hole because the Virtual I/O server NIB ping test can only test one VLAN. This leaves the other VLANs vulnerable since they are not being tested.

Since the vSwitch design permits both Virtual I/O servers network adapters to be active simultaneously, and all Virtual I/O clients validate their own network connectivity the design is able to provide full redundancy and overcome the “classic” designs' shortcomings with only one adapter in each Virtual I/O server. This makes the vSwitch design optimal because it provides the highest level of fault tolerance and flexibility at a lower price point than the “classic” SEA failover design.

Table 1 summarizes the advantages and disadvantages of the two methodologies.

Table 1: SEA Failover vs. vSwitch Comparison

	SEA Failover	**vSwitch**
Protecting against Failure of VIO Servers		
failure detection	control channel	ping
active paths to network (VIO Servers)	1	2 or more
maximum hardware usage (normal operations)	50%	100%
maximum hardware usage (failover condition)	50%	((n-1) / n * 100) where n=number of VIO servers
Protecting against Failure of Ethernet Switches, Cables or Adapters		
clients affected by failure (requred to failover)	all	some
Protecting against Misconfiguration of Virtual I/O Server		
ease of testing configuration	poor	good
Protecting against Misconfiguration of Ethernet Switch		
resiliency to traffic forwarding issues	poor	good

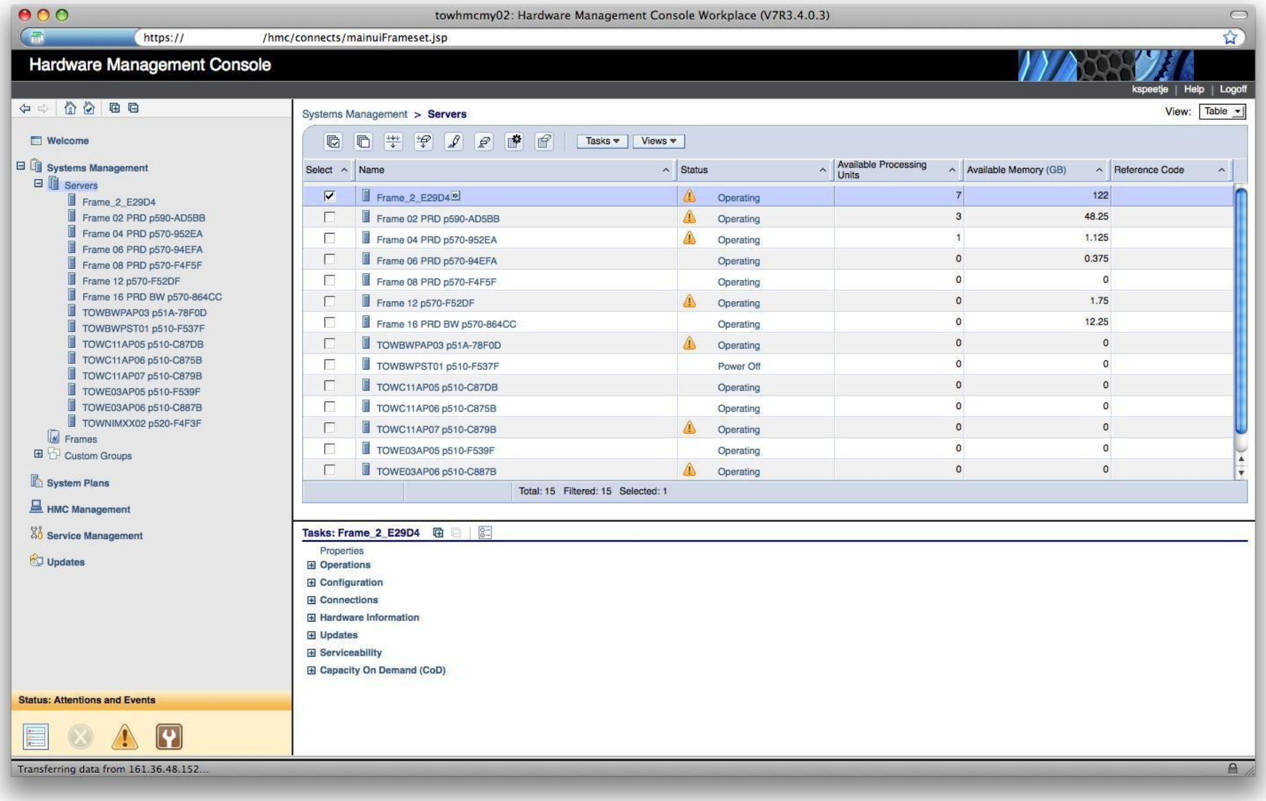
Implementation

To configure this design option, the first thing one has to do is to create a Virtual Switch (vSwitch) for each Virtual I/O Server. In this example we create vswitch1 for VIOS1 and vswitch2 for VIOS2. Order does matter or at least makes configuration a little easier. As you will see, we configure the virtual switches first, then Virtual I/O Servers, then Client LPARs. If you add virtual switches after the fact, then you will have to go into the profiles of your Virtual I/O Servers and Clients to reconfigure. This will require a restart of all affected partitions.

Part 1: Virtual Switch Creation

The first step to create the vSwitch is to log into the HMC and navigate to the area that allows you to configure the systems. An example is shown in the screen shot below.

Figure 4: Creating the virtual switch (1)



Select	Name	Status	Available Processing Units	Available Memory (GB)	Reference Code
✓	Frame_2_E29D4	Operating	7	122	
	Frame 02 PRD p590-AD5BB	Operating	3	48.25	
	Frame 04 PRD p570-952EA	Operating	1	1.125	
	Frame 06 PRD p570-94EFA	Operating	0	0.375	
	Frame 08 PRD p570-F4F5F	Operating	0	0	
	Frame 12 p570-F52DF	Operating	0	1.75	
	Frame 16 PRD BW p570-864CC	Operating	0	12.25	
	TOWBWPAP03 p51A-78F0D	Operating	0	0	
	TOWBWPST01 p510-F537F	Power Off	0	0	
	TOWC11AP05 p510-C87DB	Operating	0	0	
	TOWC11AP06 p510-C875B	Operating	0	0	
	TOWC11AP07 p510-C879B	Operating	0	0	
	TOWE03AP05 p510-F539F	Operating	0	0	
	TOWE03AP06 p510-C887B	Operating	0	0	

Total: 15 Filtered: 15 Selected: 1

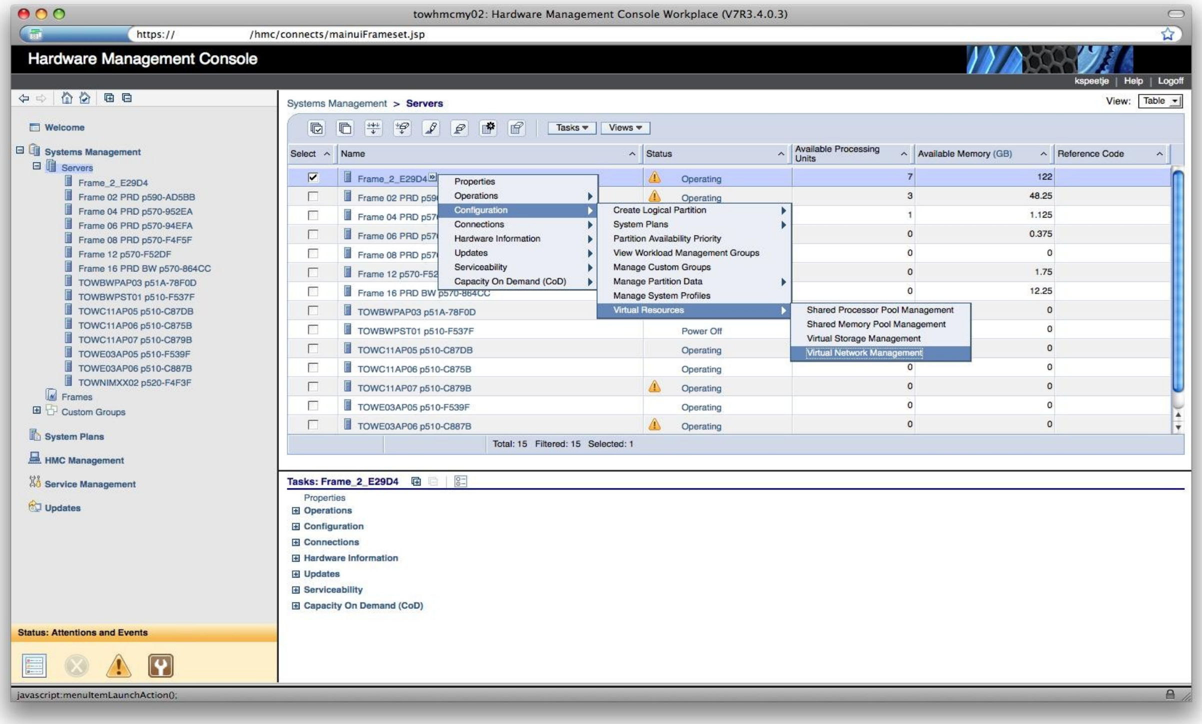
Tasks: Frame_2_E29D4

- Properties
- Operations
- Configuration
- Connections
- Hardware Information
- Updates
- Serviceability
- Capacity On Demand (CoD)

Transferring data from 161.36.48.152...

Once you are there, place a check in the box next to the Frame on which you would like to add the virtual switches. Then navigate to **Configuration -> Virtual Resources -> Virtual Network Management**.

Figure 5: Creating the virtual switch (2)

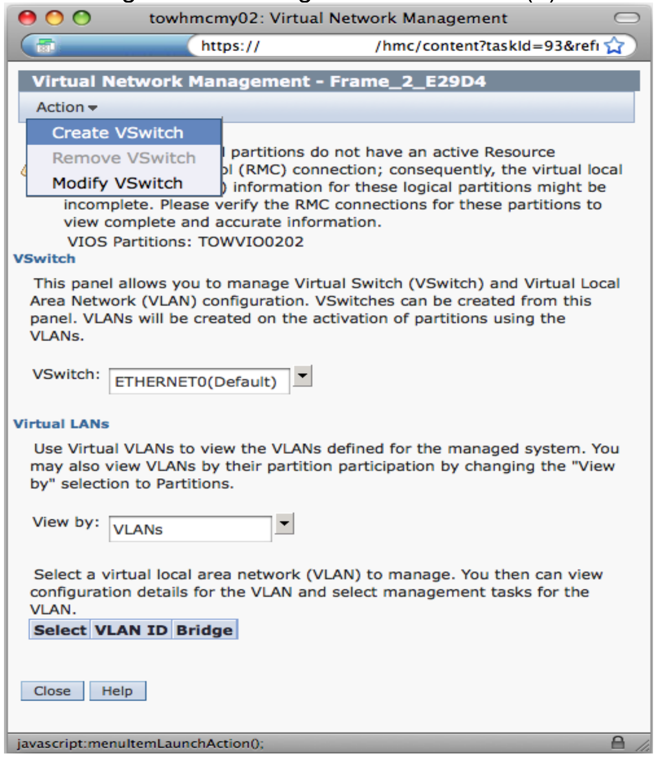


Select	Name	Status	Available Processing Units	Available Memory (GB)	Reference Code
✓	Frame_2_E29D4	Operating	7	122	
	Frame 02 PRD p59	Operating	3	48.25	
	Frame 04 PRD p57		1	1.125	
	Frame 06 PRD p57		0	0.375	
	Frame 08 PRD p57		0	0	
	Frame 12 p570-F52		0	1.75	
	Frame 16 PRD BW p570-864CC		0	12.25	
	TOWBWPAP03 p51A-78F0D			0	
	TOWBWPST01 p510-F537F	Power Off		0	
	TOWC11AP05 p510-C87DB	Operating		0	
	TOWC11AP06 p510-C875B	Operating	0	0	
	TOWC11AP07 p510-C879B	Operating	0	0	
	TOWE03AP05 p510-F539F	Operating	0	0	
	TOWE03AP06 p510-C887B	Operating	0	0	

Total: 15 Filtered: 15 Selected: 1

The following screen appears. Notice the default Ethernet switch (ETHERNET0). You will not be using the default switch in the environment so be sure to avoid it when setting up networking.

Figure 6: Creating the virtual switch (3)



towhmcmy02: Virtual Network Management
https:// /hmc/content?taskId=93&refi
Action
Create VSwitch
Remove VSwitch
Modify VSwitch
Close
Help
javascript:menuItemLaunchAction();

Virtual Network Management - Frame_2_E29D4

l partitions do not have an active Resource
ol (RMC) connection; consequently, the virtual local
) information for these logical partitions might be
incomplete. Please verify the RMC connections for these partitions to
view complete and accurate information.
VIOS Partitions: TOWVIO0202

VSwitch

This panel allows you to manage Virtual Switch (VSwitch) and Virtual Local Area Network (VLAN) configuration. VSwitches can be created from this panel. VLANs will be created on the activation of partitions using the VLANs.

VSwitch: ETHERNET0(Default)

Virtual LANs

Use Virtual VLANs to view the VLANs defined for the managed system. You may also view VLANs by their partition participation by changing the "View by" selection to Partitions.

View by: VLANs

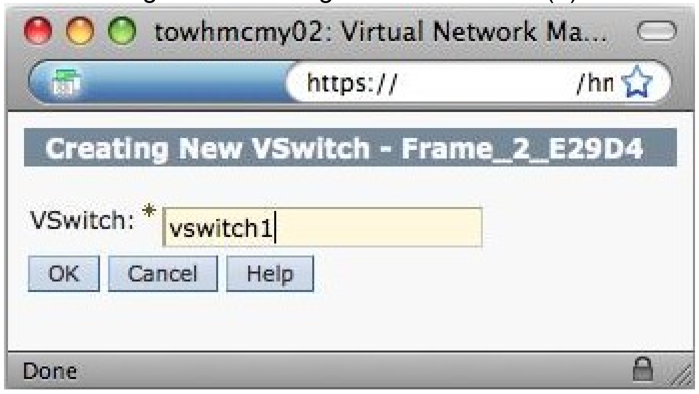
Select a virtual local area network (VLAN) to manage. You then can view configuration details for the VLAN and select management tasks for the VLAN.

Select	VLAN ID	Bridge

Next click the **Action** button at the top right of the screen and choose to "**Create VSwitch**"

The following screen appears and allows you to enter a name for the new virtual switch, in this case, "**vswitch1**". Then click "**OK**".

Figure 7: Creating the virtual switch (4)



towhmcmy02: Virtual Network Ma...

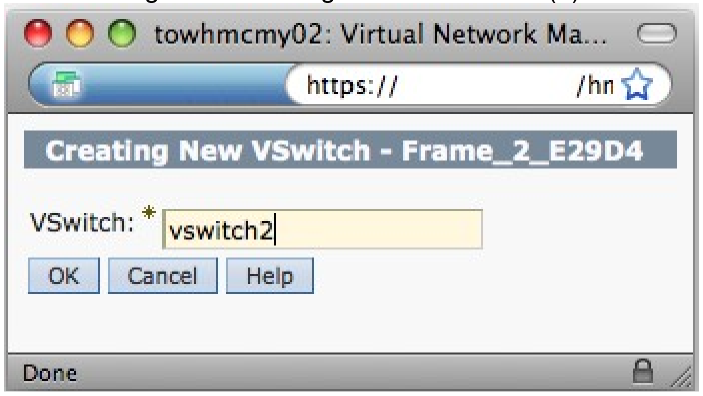
https:// /hn

Creating New VSwitch - Frame_2_E29D4

VSwitch: * vswitch1

Repeat the previous step to create vswitch2.

Figure 8: Creating the virtual switch (5)



towhmcmy02: Virtual Network Ma...

https:// /hn

Creating New VSwitch - Frame_2_E29D4

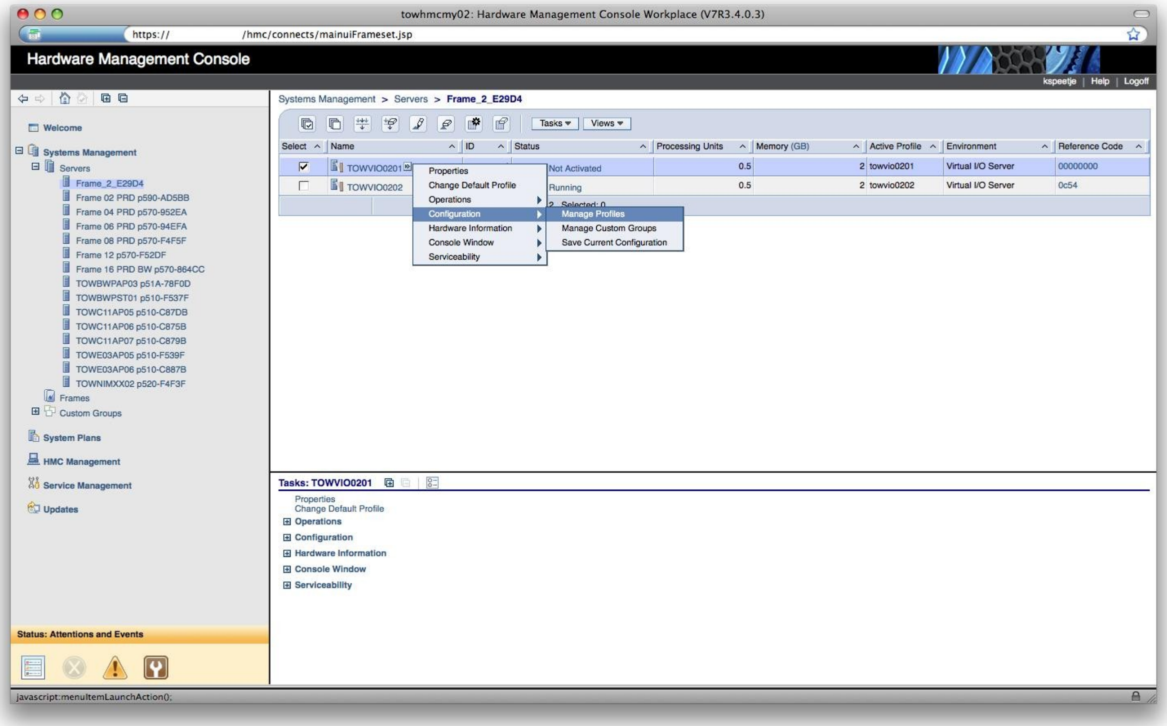
VSwitch: * vswitch2

Now that the two virtual switches have been created it is time to create the Virtual I/O Server and client configurations to allow traffic to be bridged.

Part 2: Virtual Ethernet Server Adapter

At this point the Virtual I/O Server has been created, and we are editing the profile to create the virtual Ethernet server adapter on the Virtual I/O Server.

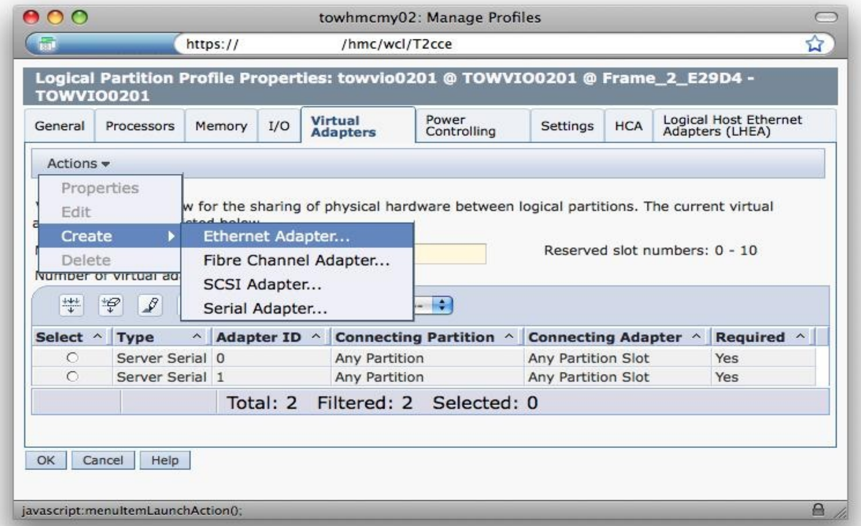
Figure 9: Creating the virtual Ethernet server adapter (1)


towhmcmy02: Hardware Management Console Workplace (V7R3.4.0.3)
https:// /hmc/connects/mainuiFrameset.jsp
Hardware Management Console
kspeetje | Help | Logoff
Systems Management > Servers > Frame_2_E29D4
Tasks
Views
Properties
Change Default Profile
Operations
Configuration
Hardware Information
Console Window
Serviceability
Manage Profiles
Manage Custom Groups
Save Current Configuration
Tasks: TOWVIO0201
javascript:menuItemLaunchAction();

Select	Name	ID	Status	Processing Units	Memory (GB)	Active Profile	Environment	Reference Code
✓	TOWVIO0201		Not Activated	0.5	2	towvio0201	Virtual I/O Server	00000000
	TOWVIO0202		Running	0.5	2	towvio0202	Virtual I/O Server	0c54

Click on the **Virtual Adapters** tab, then click **Actions → Create → Ethernet Adapter**.

Figure 10: Creating the virtual Ethernet server adapter (2)


towhmcmy02: Manage Profiles
https:// /hmc/wcl/T2cce
Logical Partition Profile Properties: towvio0201 @ TOWVIO0201 @ Frame_2_E29D4 - TOWVIO0201
General
Processors
Memory
I/O
Virtual Adapters
Power Controlling
Settings
HCA
Logical Host Ethernet Adapters (LHEA)
Actions
Properties
Edit
Create
Delete
Ethernet Adapter...
Fibre Channel Adapter...
SCSI Adapter...
Serial Adapter...
Reserved slot numbers: 0 - 10
OK
Cancel
Help
javascript:menuItemLaunchAction();

Select	Type	Adapter ID	Connecting Partition	Connecting Adapter	Required
○	Server Serial	0	Any Partition	Any Partition Slot	Yes
○	Server Serial	1	Any Partition	Any Partition Slot	Yes

Total: 2   Filtered: 2   Selected: 0

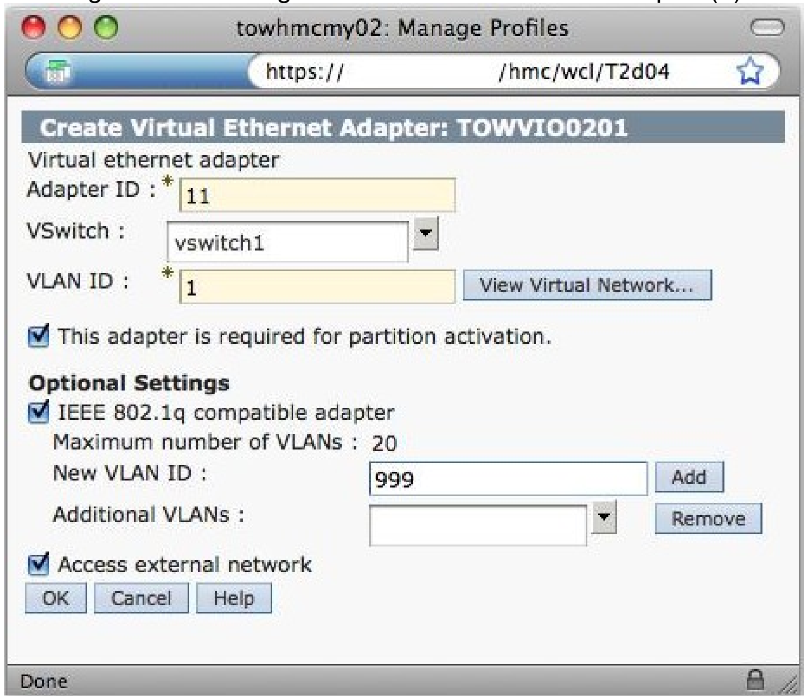
The first blank is the Adapter ID. Here it is set to 11. For consistency, our Virtual I/O Servers all have their first virtual Ethernet adapter in slot 11.

The next blank is the VSwitch. Since this is VIOS1, we are "plugging it into" vswitch1 by selecting that virtual switch. The next step is to configure the VLAN ID. Since this adapter will carry several VLANs, the VLAN ID blank is for the most part irrelevant.

Since this adapter has to carry multiple VLANs, it has to be an "**IEEE 802.1q compatible adapter**". That's a fancy way of saying it has to allow VLAN tagged packets.

The next blanks are for defining which ports are allowed into the switch from the virtual hosts. In our environment, we configured VLANs 999, 46, 47, 48, 49 and 50.

Figure 11: Creating the virtual Ethernet server adapter (3)



towhmcmy02: Manage Profiles

https:// /hmc/wcl/T2d04

Create Virtual Ethernet Adapter: TOWVIO0201

Virtual ethernet adapter

Adapter ID :* 11

VSwitch : vswitch1

VLAN ID : * 1

☑ This adapter is required for partition activation.

Optional Settings

☑ IEEE 802.1q compatible adapter

Maximum number of VLANs : 20

New VLAN ID : 999

Additional VLANs :

☑ Access external network

Done

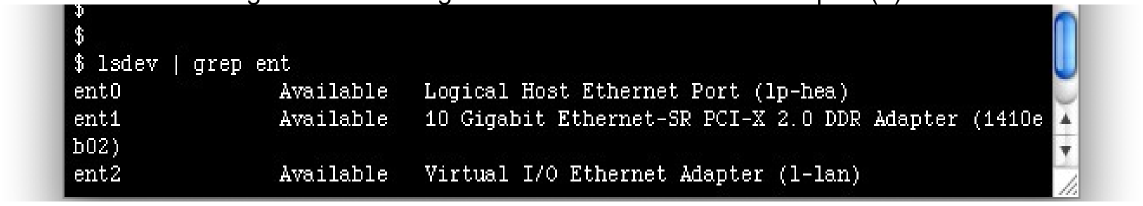
The last step is to check the "**Access external network**". This step is what gives you the ability later on to define that this adapter can be used as part of a SEA.

Once this is complete, click "**OK**" and exit to the main screen of the HMC. Then repeat the steps for VIOS2, with the only change being is that you will now specify vswitch2 for the Virtual Switch. It's important that the two adapters are exactly the same to allow for failover of the client LPAR network.

This step has defined to the switches which VLANs can enter into the switch. At this point you could attach LPARs to the switch and allow them to communicate between each other. However, since you haven't tied the virtual and physical together, the LPARs are not able to reach systems outside of the frame. In part 3, we'll be creating the Shared Ethernet adapter which connects the virtual to the physical worlds through the Virtual I/O Servers.

However, before you can do that, power on the Virtual I/O Server LPARs, log in as padmin and verify that the configuration is good thus far. Start by typing in the command **lsdev | grep ent**.

Figure 12: Creating the virtual Ethernet server adapter (4)

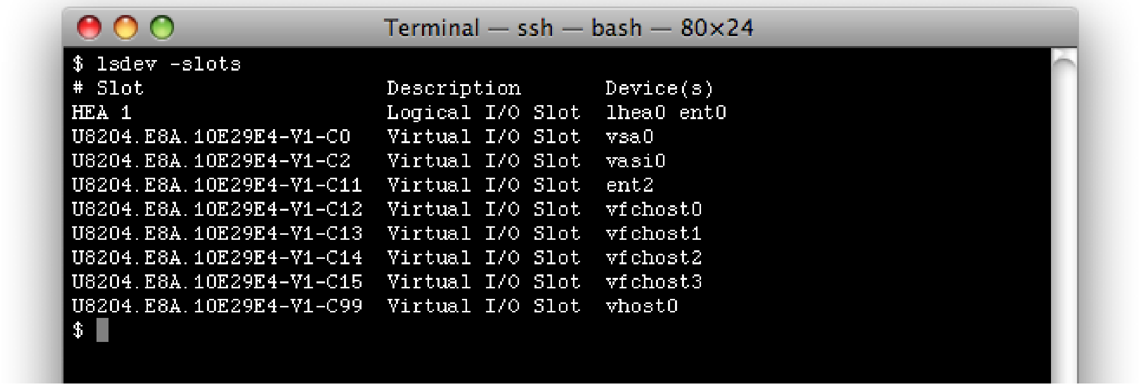


```
$
$
$ lsdev | grep ent
ent0             Available   Logical Host Ethernet Port (lp-hea)
ent1             Available   10 Gigabit Ethernet-SR PCI-X 2.0 DDR Adapter (1410e
b02)
ent2             Available   Virtual I/O Ethernet Adapter (l-lan)
```

You can see in this screen shot that ent2 is the Virtual I/O Ethernet Adapter that was created in the previous steps.

You can verify that ent2 is in slot 11 by typing **lsdev –slots**. This can also be used to correlate all virtual devices created in the HMC interface to the AIX adapters created when the configuration manager (cfgmgr) is run.

Figure 13: Creating the virtual Ethernet server adapter (5)



```
$ lsdev -slots
# Slot                     Description       Device(s)
HEA 1                      Logical I/O Slot  lhea0 ent0
U8204.E8A.10E29E4-V1-C0    Virtual I/O Slot  vsa0
U8204.E8A.10E29E4-V1-C2    Virtual I/O Slot  vasi0
U8204.E8A.10E29E4-V1-C11   Virtual I/O Slot  ent2
U8204.E8A.10E29E4-V1-C12   Virtual I/O Slot  vfchost0
U8204.E8A.10E29E4-V1-C13   Virtual I/O Slot  vfchost1
U8204.E8A.10E29E4-V1-C14   Virtual I/O Slot  vfchost2
U8204.E8A.10E29E4-V1-C15   Virtual I/O Slot  vfchost3
U8204.E8A.10E29E4-V1-C99   Virtual I/O Slot  vhost0
$ 
```

One thing to note when configuring network on the Virtual I/O Server, it is important that the network interfaces are kept consistent from environment to environment. In our example environment, the LHEA adapter is always ent0 and the 10Gb/s adapter is ent1. If they do not come up this way when you initially boot the system, you will need to perform the following commands.

rmdev –dl et0
rmdev –dl et1
rmdev –dl en0
rmdev –dl en1
rmdev –dl ent0
rmdev –dl ent1

Shut down the LPAR and remove the 10Gb/s adapter from the profile. Reboot the system so that the LHEA will be ent0. You then shutdown the LPAR again, add the 10Gb/s adapter to the profile and boot it one more time. It's a little extra work that will help everything make a lot more sense later on in the process.

Part 3: Shared Ethernet Adapter

The next process is to create the Shared Ethernet Adapter. This step is what effectively ties the virtual host adapters, the virtual switch, and the physical interface together. The command is:
mkvdev –sea ent1 –vadapter ent2 –default ent2 –defaultid 1

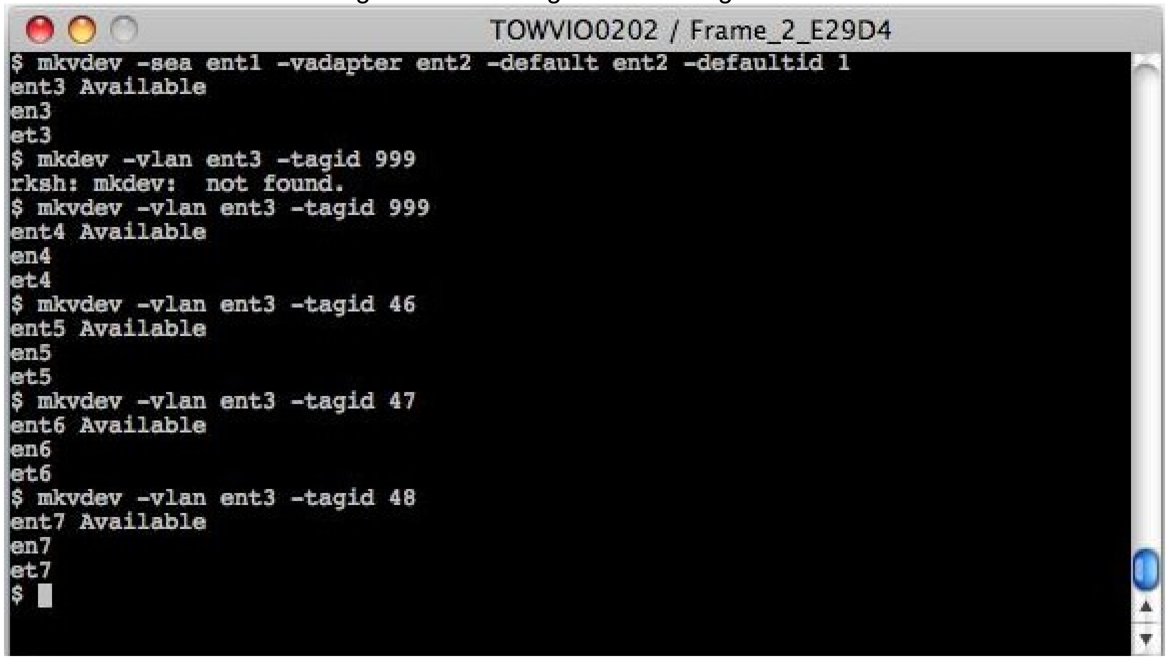
This command breaks down into two parts. The **mkvdev –sea ent1 –vadapter ent2** part means that you are connecting the physical device ent1 (the 10Gb/s adapter) to the virtual adapter you just created. The **-default ent2 –defaultid 1** part indicates that ent2 is to be used as the adapter to send untagged packets out and tag them with the default VLAN of 1 as defined by the –defaultid.

In our case, the VLAN of 1 was given to us by the network team as the native VLAN. By setting this value, we are complying with their configuration. Since VLAN 1 isn't used by systems in the this network what the command effectively does is throw a packet away if or when something sends the packet without a tag. This is common in environments where VLAN tagging is used to separate traffic.

The last step is to allow the VLANS that were defined while making the virtual interface to be passed through the SEA to the physical interface. The command to accomplish this goal is:
mkvdev –vlan ent3 –tagid *XXX*

Below is a screen shot of both the SEA and VLANs being defined.

Figure 14: Creating the VLAN Tag Devices



TOWVIO0202 / Frame_2_E29D4

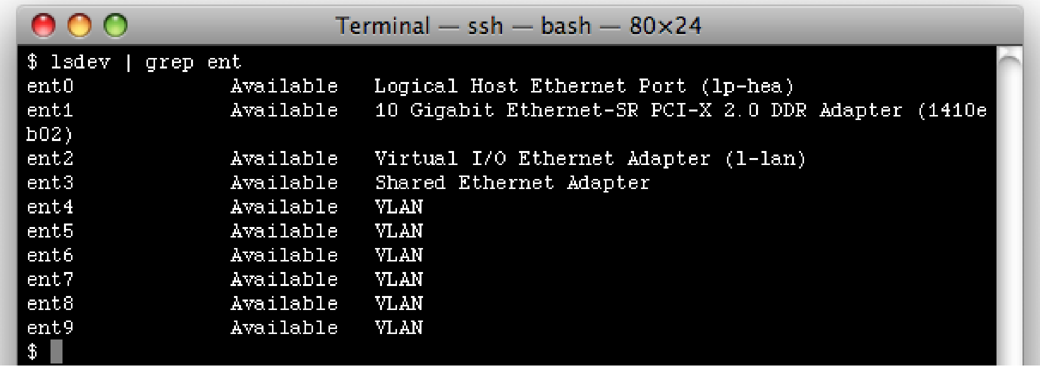
```
$ mkvdev -sea ent1 -vadapter ent2 -default ent2 -defaultid 1
ent3 Available
en3
et3
$ mkdev -vlan ent3 -tagid 999
rksh: mkdev:  not found.
$ mkvdev -vlan ent3 -tagid 999
ent4 Available
en4
et4
$ mkvdev -vlan ent3 -tagid 46
ent5 Available
en5
et5
$ mkvdev -vlan ent3 -tagid 47
ent6 Available
en6
et6
$ mkvdev -vlan ent3 -tagid 48
ent7 Available
en7
et7
$
```

Note the order in which the VLANs are created. It is important that they are consistent from environment to environment. In our case, the order was 999, 46 ,47, 48, 49 and 50 and they should correspond to:

VLAN	Adapter
999	ent4
46	ent5
47	ent6
48	ent7
49	ent8
50	ent9

When finished the output of the command **lsdev | grep ent** should look like the following screen shot:

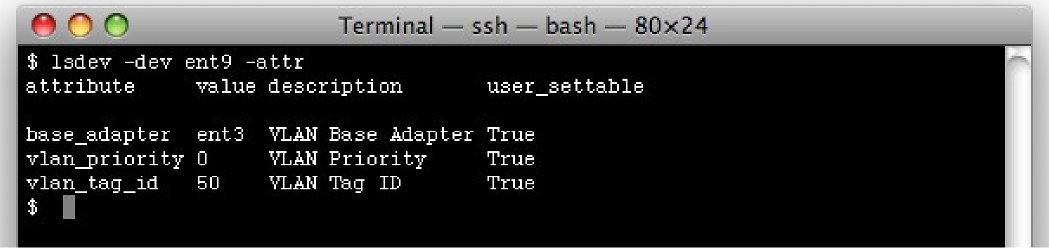
Figure 15: Viewing the VLAN Tag Devices



```
$ lsdev | grep ent
ent0             Available   Logical Host Ethernet Port (lp-hea)
ent1             Available   10 Gigabit Ethernet-SR PCI-X 2.0 DDR Adapter (1410e
b02)
ent2             Available   Virtual I/O Ethernet Adapter (l-lan)
ent3             Available   Shared Ethernet Adapter
ent4             Available   VLAN
ent5             Available   VLAN
ent6             Available   VLAN
ent7             Available   VLAN
ent8             Available   VLAN
ent9             Available   VLAN
$
```

If you wish to find out what VLAN an adapter represents, the command is **lsdev –dev ent*X* –attr.**

Figure 16: Viewing the VLAN Tag Attributes



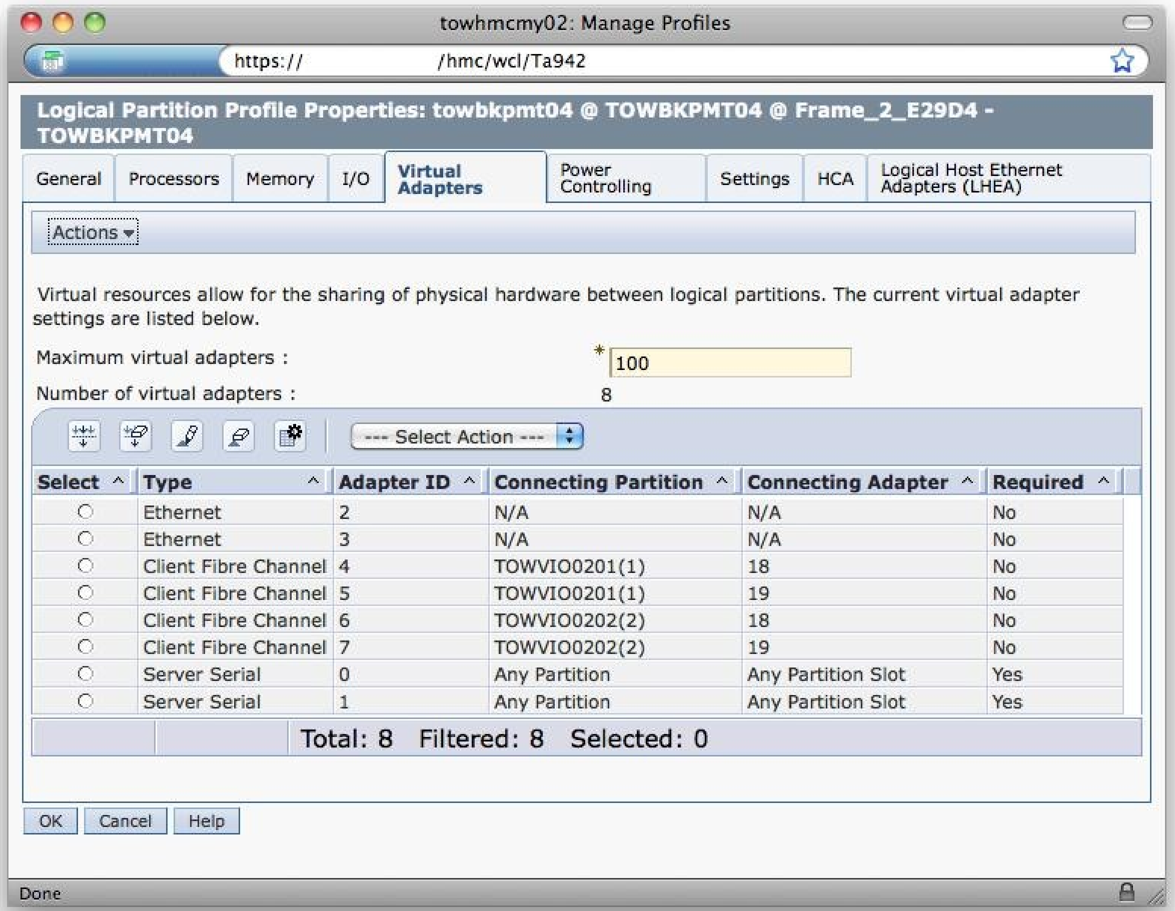
```
$ lsdev -dev ent9 -attr
attribute     value description       user_settable

base_adapter  ent3  VLAN Base Adapter True
vlan_priority 0     VLAN Priority     True
vlan_tag_id   50    VLAN Tag ID       True
$
```

Part 4: Virtual Client Network Interface

The next process is to set up the Virtual Ethernet Client adapters on the client LPARs.

The first step is to create the adapters in the LPAR profile. Every LPAR should have two Ethernet adapters. Again, for consistency sake, we configured one in slot 2 and one in slot 3.

Figure 17: Viewing the Client's Virtual Adapters



towhmcmy02: Manage Profiles

https:// /hmc/wcl/Ta942

Logical Partition Profile Properties: towbkpmt04 @ TOWBKPMT04 @ Frame_2_E29D4 - TOWBKPMT04

General | Processors | Memory | I/O | Virtual Adapters | Power Controlling | Settings | HCA | Logical Host Ethernet Adapters (LHEA)

Actions

Virtual resources allow for the sharing of physical hardware between logical partitions. The current virtual adapter settings are listed below.

Maximum virtual adapters : * 100

Number of virtual adapters : 8

--- Select Action ---

Select	Type	Adapter ID	Connecting Partition	Connecting Adapter	Required
○	Ethernet	2	N/A	N/A	No
○	Ethernet	3	N/A	N/A	No
○	Client Fibre Channel	4	TOWVIO0201(1)	18	No
○	Client Fibre Channel	5	TOWVIO0201(1)	19	No
○	Client Fibre Channel	6	TOWVIO0202(2)	18	No
○	Client Fibre Channel	7	TOWVIO0202(2)	19	No
○	Server Serial	0	Any Partition	Any Partition Slot	Yes
○	Server Serial	1	Any Partition	Any Partition Slot	Yes

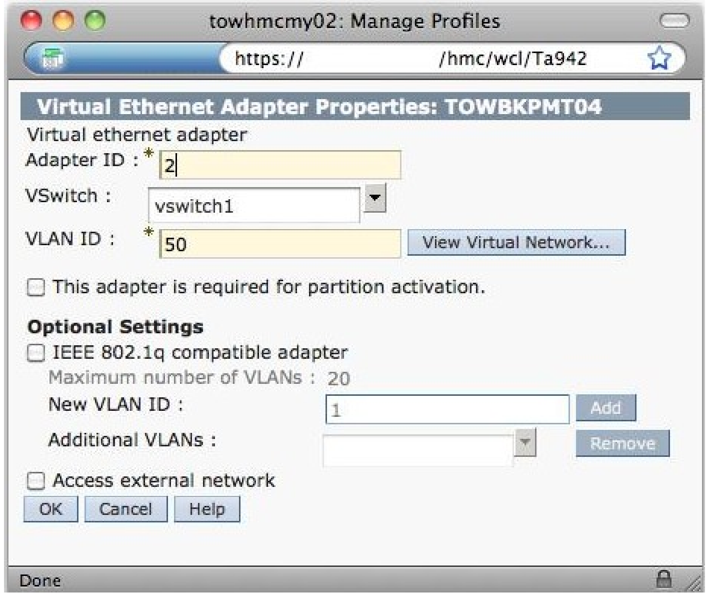
Total: 8 Filtered: 8 Selected: 0

OK Cancel Help

Done

The one in slot 2 should look similar to the following.

Figure 18: Configuring the Client's First Virtual Ethernet Adapter Properties



towhmcmy02: Manage Profiles

https:// /hmc/wcl/Ta942

Virtual Ethernet Adapter Properties: TOWBKPMT04

Virtual ethernet adapter

Adapter ID : * 2

VSwitch : vswitch1

VLAN ID : * 50 View Virtual Network...

☐ This adapter is required for partition activation.

Optional Settings

☐ IEEE 802.1q compatible adapter

Maximum number of VLANs : 20

New VLAN ID : 1 Add

Additional VLANs : Remove

☐ Access external network

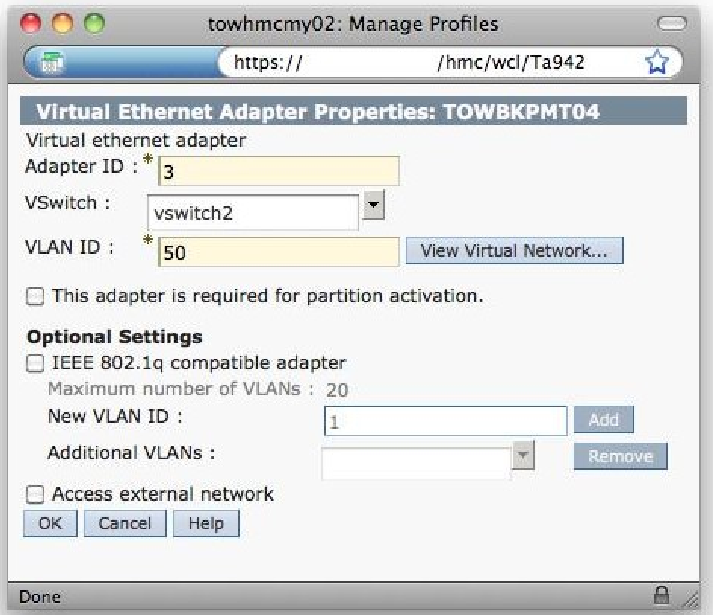
OK Cancel Help

Done

The important items in this adapter configuration are the **VSwitch**, the **VLAN ID**, and the fact that this adapter is **not** required for partition activation. (If you check the “This adapter is required for partition activation” you cannot use this LPAR for live partition mobility.)

The adapter in slot 3 must be identical to the adapter in slot 2 except for the fact that it is “plugged into” vswitch2.

Figure 19: Configuring the Client's Second Virtual Ethernet Adapter Properties



towhmcmy02: Manage Profiles

https:// /hmc/wcl/Ta942

Virtual Ethernet Adapter Properties: TOWBKPMT04

Virtual ethernet adapter

Adapter ID : * 3

VSwitch : vswitch2

VLAN ID : * 50

☐ This adapter is required for partition activation.

Optional Settings

☐ IEEE 802.1q compatible adapter

Maximum number of VLANs : 20

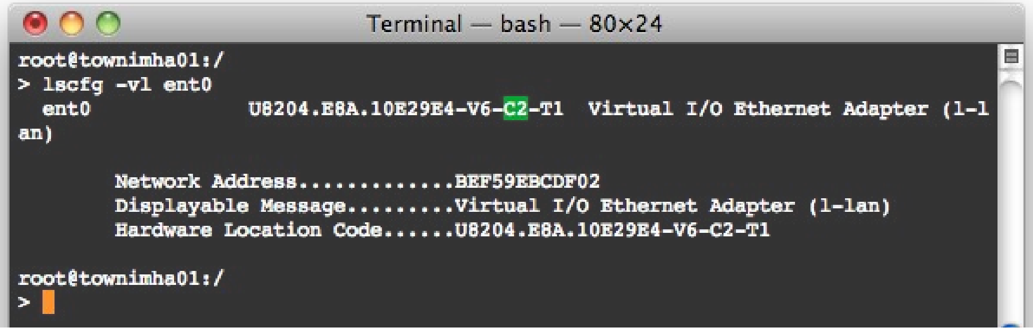
New VLAN ID : 1

Additional VLANs :

☐ Access external network

After installation you can verify that ent0 maps to slot 2 by issuing the command l**scfg –vl ent0**. The highlighted C2 below shows that the ent0 adapter is in slot 2.

Figure 20: Checking configuration of Virtual Ethernet Adapter

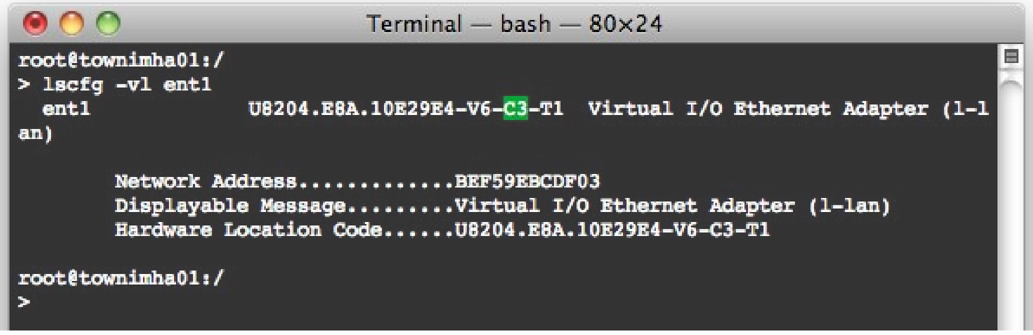


Terminal — bash — 80×24

```
root@townimha01:/
> lscfg -vl ent0
  ent0             U8204.E8A.10E29E4-V6-C2-T1  Virtual I/O Ethernet Adapter (l-l
an)

        Network Address.............BEF59EBCDF02
        Displayable Message.........Virtual I/O Ethernet Adapter (l-lan)
        Hardware Location Code......U8204.E8A.10E29E4-V6-C2-T1

root@townimha01:/
> 
```

Issuing the same command on ent1 gives the following result.

Figure 21: Checking configuration of Virtual Ethernet Adapter



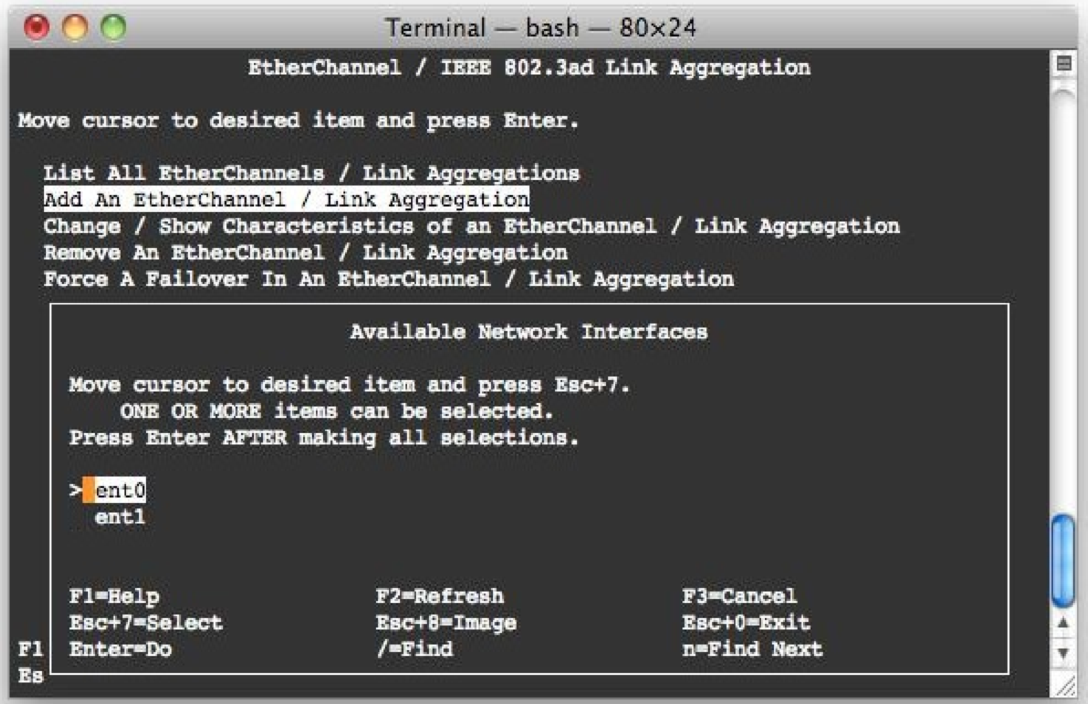
```
root@townimha01:/
> lscfg -vl ent1
  ent1             U8204.E8A.10E29E4-V6-C3-T1  Virtual I/O Ethernet Adapter (l-l
an)

        Network Address.............BEF59EBCDF03
        Displayable Message.........Virtual I/O Ethernet Adapter (l-lan)
        Hardware Location Code......U8204.E8A.10E29E4-V6-C3-T1

root@townimha01:/
>
```

The next step is to create the Network Interface Backup (NIB) device. To do this enter the command **smitty etherchannel**.

In the smit screen that appears chose the option that says **Add An EtherChannel / Link Aggregation.**

Select the adapters that will be part of the Etherchannel.

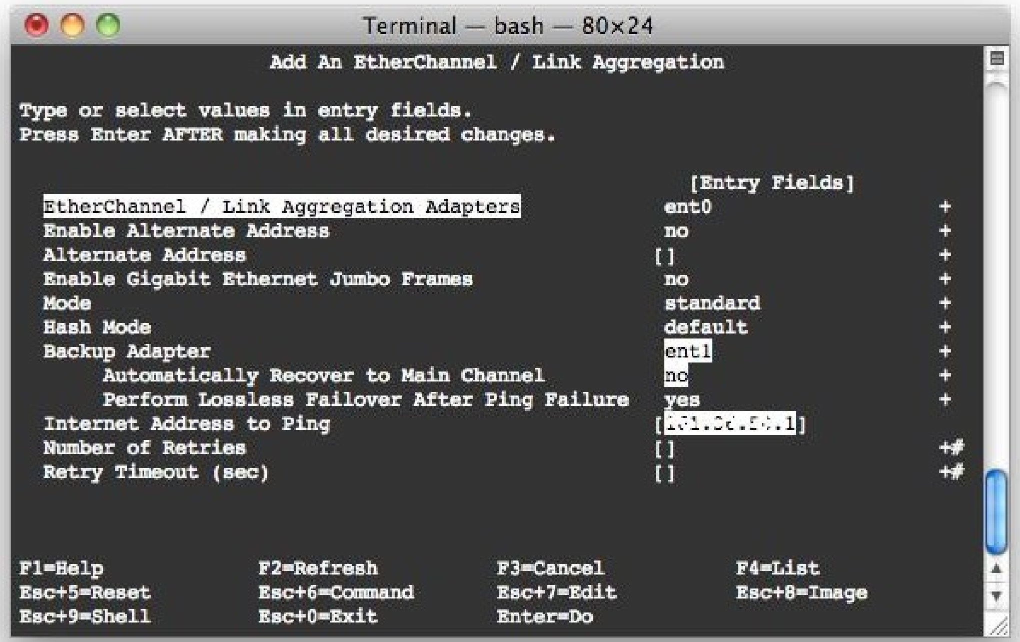
Figure 22: Configuring the NIB device (1)



```
                  EtherChannel / IEEE 802.3ad Link Aggregation

Move cursor to desired item and press Enter.

  List All EtherChannels / Link Aggregations
  Add An EtherChannel / Link Aggregation
  Change / Show Characteristics of an EtherChannel / Link Aggregation
  Remove An EtherChannel / Link Aggregation
  Force A Failover In An EtherChannel / Link Aggregation
  +-------------------------------------------------------------------------+
  |                      Available Network Interfaces                       |
  |                                                                         |
  | Move cursor to desired item and press Esc+7.                            |
  |     ONE OR MORE items can be selected.                                  |
  | Press Enter AFTER making all selections.                                |
  |                                                                         |
  | > ent0                                                                  |
  |   ent1                                                                  |
  |                                                                         |
  |                                                                         |
  | F1=Help                 F2=Refresh              F3=Cancel               |
  | Esc+7=Select            Esc+8=Image             Esc+0=Exit              |
F1| Enter=Do                /=Find                  n=Find Next             |
Es+-------------------------------------------------------------------------+
```

In the screen that follows, add **ent1** as a backup adapter, change the Automatically Recover to Main Channel to **no**, and set the Internet IP address to ping to be the default GW for the server.

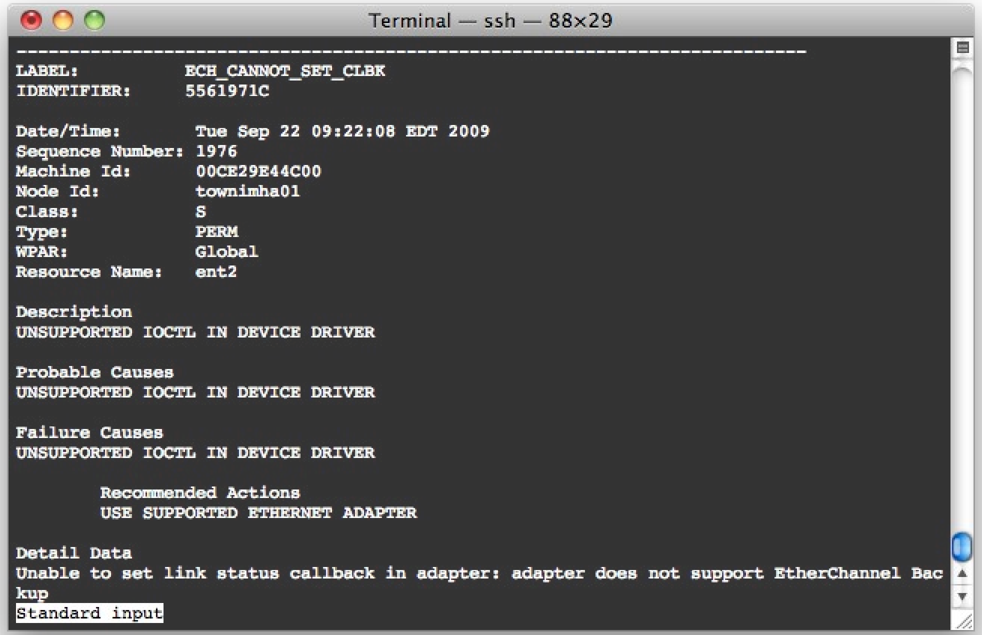
Figure 23: Configuring the NIB device (2)



```
                      Add An EtherChannel / Link Aggregation

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                         [Entry Fields]
  EtherChannel / Link Aggregation Adapters             ent0                    +
  Enable Alternate Address                             no                      +
  Alternate Address                                   []                       +
  Enable Gigabit Ethernet Jumbo Frames                 no                      +
  Mode                                                 standard                +
  Hash Mode                                            default                 +
  Backup Adapter                                       ent1                    +
       Automatically Recover to Main Channel           no                      +
       Perform Lossless Failover After Ping Failure    yes                     +
  Internet Address to Ping                            [[redacted]]
  Number of Retries                                   []                       +#
  Retry Timeout (sec)                                 []                       +#


F1=Help             F2=Refresh          F3=Cancel           F4=List
Esc+5=Reset         Esc+6=Command       Esc+7=Edit          Esc+8=Image
Esc+9=Shell         Esc+0=Exit          Enter=Do
```

After configuring the Etherchannel device, you will see the following error in the errpt output. You can ignore the error because it basically says a virtual adapter can't see a status change of a physical connection. This is true of course, but we compensate for that by pinging the default gateway.

Figure 24: Examining the Error Log



```
---------------------------------------------------------------------------
LABEL:          ECH_CANNOT_SET_CLBK
IDENTIFIER:     5561971C

Date/Time:       Tue Sep 22 09:22:08 EDT 2009
Sequence Number: 1976
Machine Id:      00CE29E44C00
Node Id:         townimha01
Class:           S
Type:            PERM
WPAR:            Global
Resource Name:   ent2

Description
UNSUPPORTED IOCTL IN DEVICE DRIVER

Probable Causes
UNSUPPORTED IOCTL IN DEVICE DRIVER

Failure Causes
UNSUPPORTED IOCTL IN DEVICE DRIVER

        Recommended Actions
        USE SUPPORTED ETHERNET ADAPTER

Detail Data
Unable to set link status callback in adapter: adapter does not support EtherChannel Bac
kup
Standard input
```

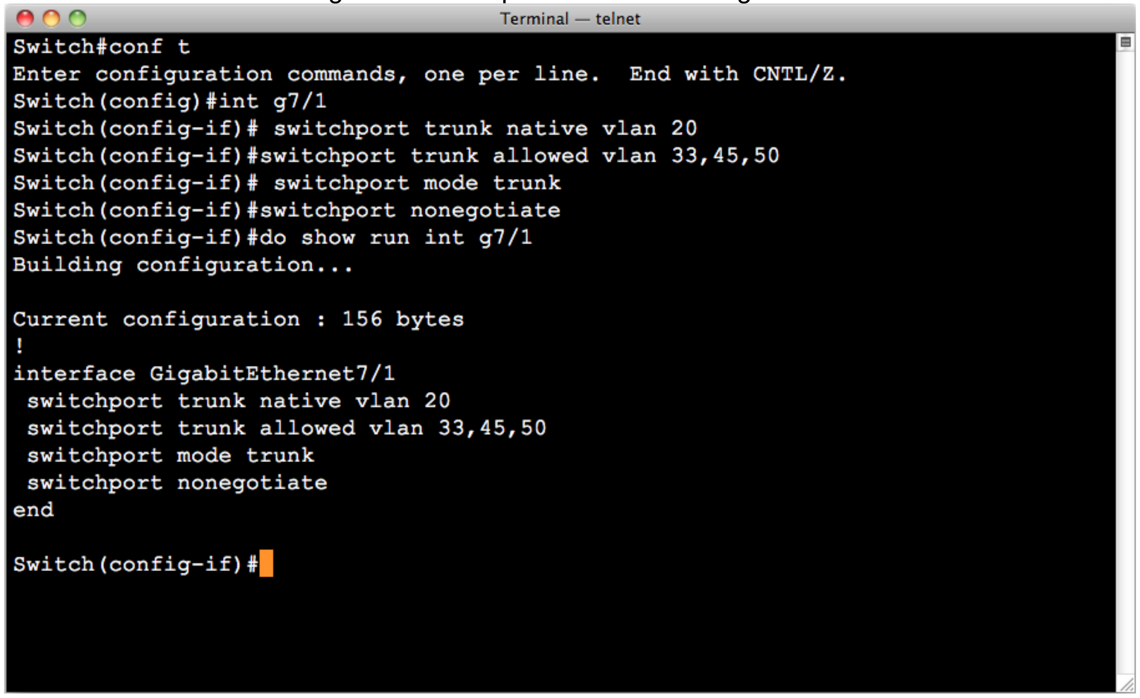
The last step is to configure an IP address on the newly created Etherchannel interface. You do this just like every other device with the command **smitty tcpip**. The most important piece here is that the IP address has to be on the VLAN that was assigned to the adapters in the previous steps. You also have to remember to change the VLAN for the interfaces if you want to move this system to a different VLAN.

Part 5: Cisco Switch Configuration

The switches that were used in our implementation happened to be Cisco products. We have added this section for completeness. It is not to imply any preference toward any particular vendor of switch products.

Configuration of the switch to support the previous example is very simple to setup, maintain, and secure. The example below shows a Cisco 4507E 1 Gb/s switch port configured as a trunk (a 10 Gb/s port would be nearly identical). The command **switchport mode trunk** configures the port to behave as a trunk. The command **switchport native vlan 20** established which VLAN is used to tag packets that arrive at the interface without a tag. It is also used during the initial establishment of the trunk so it is important that it matches the **-defaultid** of the SEA. The network administrator can control which VLANs are allowed to traverse the trunk with the command **switchport trunk allowed vlan 33,45,50**. The **switchport nonegotiate** command stops Dynamic Trunking Port (DTP) packets from being sent.

Figure 25: Example Cisco Port Configuration

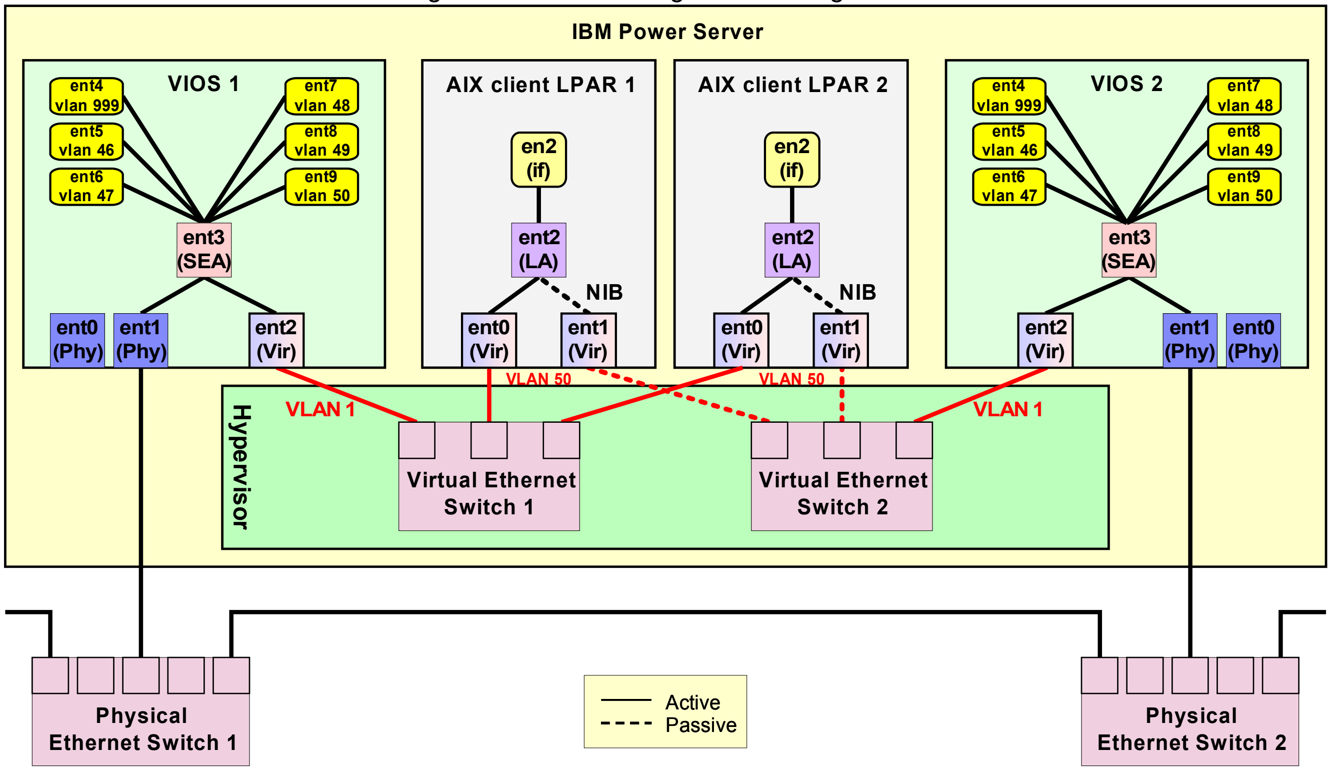


```
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int g7/1
Switch(config-if)# switchport trunk native vlan 20
Switch(config-if)#switchport trunk allowed vlan 33,45,50
Switch(config-if)# switchport mode trunk
Switch(config-if)#switchport nonegotiate
Switch(config-if)#do show run int g7/1
Building configuration...

Current configuration : 156 bytes
!
interface GigabitEthernet7/1
 switchport trunk native vlan 20
 switchport trunk allowed vlan 33,45,50
 switchport mode trunk
 switchport nonegotiate
end

Switch(config-if)#
```

Conclusion

Here is one last diagram of how the configuration looks "as-built".

Figure 26: As-Built Diagram of Configuration



The use of Virtual Switches can provide an important new design choice when implementing a Power Server particularly if your infrastructure uses VLAN tagging. We hope that this paper has helped you understand the value and benefits that using Virtual Switches can offer and that you can consider it when planning your next implementation. Additionally, we hope that the Implementation section provides clear and sufficiently detailed guidance on the steps required to configure a Power System with Virtual Switches as well as the additional configuration steps in the Virtual I/O Servers and Clients.

References

Here is a link to a good document that explains the concept, benefits and configuration of VLANs, particularly geared toward its use with the Virtual I/O Server. Entitled “POWER5 Virtualization: How to work with VLANs using the IBM Virtual I/O Server”, it is still applicable to POWER6 and POWER7.

http://www.ibm.com/developerworks/systems/library/es-pwr5-virtualvlan/index.html